From Sacred Taboos to AI Jailbreaks: The History of Illegal Numbers, Censored Codes, Cryptographic Keys, and Executable Information

Douglas C. Youvan

doug@youvan.com www.youvan.ai

June 19, 2026

Numbers are often treated as the purest objects of reason: abstract, universal, and politically neutral. Yet throughout history, particular numbers, symbols, dates, codes, and sequences have been feared, censored, restricted, or criminalized because of what they signify or what they enable. Ancient communities guarded numerical knowledge that threatened religious doctrine, Roman authorities treated calculations of an emperor's death as treasonous, and medieval merchants distrusted positional notation because it could facilitate fraud. In the modern era, telephone tones, passwords, product keys, encryption keys, software encoded as prime numbers, cryptocurrency addresses, and adversarial AI strings transformed numerical expressions from passive representations into operational instruments. A sequence could now unlock a network, decrypt a film, authorize a financial transfer, exploit software, evade censorship, or manipulate a machine-learning system. This paper traces the history of "illegal numbers" from antiquity to the AI and synthetic-biology frontier. It argues that numbers are rarely prohibited as mathematical entities; rather, law and power regulate their representation, circulation, intended use, and causal capacity. The history reveals a widening convergence between speech, software, identity, property, threat, and executable information.

Keywords: illegal numbers, censored codes, cryptographic keys, executable information, DeCSS, illegal prime numbers, AACS encryption key, DMCA, source code, free speech, numerical taboo, Pythagoreans, square root of two, Hebrew numerals, Roman treason, zero, telephone phreaking, 2600 Hz, political codes, cryptocurrency sanctions, adversarial AI, jailbreak tokens, synthetic biology, information hazards.

OpenAI GPT-5.5 Thinking. CC4.0. 128 pages.

https://orcid.org/0009-0004-4206-4253

Outline

1. Sacred Numbers, Forbidden Knowledge, and the Ancient Origins of Numerical Control

1.1 Numbers as Religious and Cosmological Principles

The ancient belief that numerical relationships revealed the fundamental structure of reality.

1.2 The Pythagorean Worldview

The role of whole numbers, ratios, harmony, music, geometry, and cosmic order in Pythagorean thought.

1.3 The Irrationality of the Square Root of Two

Why the diagonal of a unit square presented a profound challenge to a worldview founded on whole-number ratios.

1.4 Hippasus and the Legend of the Drowned Mathematician

The traditional account of punishment for revealing irrational magnitudes, its historical uncertainty, and its later symbolic importance.

1.5 Secrecy Versus Statutory Illegality

The difference between restricted knowledge inside a philosophical community and a number prohibited by formal law.

1.6 Numerical Knowledge as a Threat to Authority

How mathematical discoveries can destabilize institutions whose legitimacy depends on a particular account of reality.

2. Sacred Writing, Numerical Substitution, and the Hebrew Treatment of Fifteen and Sixteen

2.1 Letters as Numbers

The operation of alphabetic numeral systems in which written symbols simultaneously carry linguistic and numerical meanings.

2.2 The Divine Name and Reverential Avoidance

Why the expected Hebrew representations of fifteen and sixteen resemble abbreviated forms of the sacred name.

2.3 Nine Plus Six and Nine Plus Seven

The substitution of alternative numerical combinations while preserving the underlying quantities.

2.4 Numerical Notation as Sacred Language

How the same marks can function as arithmetic symbols, words, names, and religious objects.

2.5 Taboo Without Criminalization

Why reverential conventions should not be confused with modern statutory prohibitions.

2.6 The Persistence of Ancient Numerical Conventions

The continued use of these forms in Hebrew dates, calendars, page numbering, and religious writing.

3. Forbidden Predictions, Imperial Death Dates, and Numerical Treason in Rome

3.1 Astrology as Political Intelligence

Why astronomical and astrological calculations could be treated as more than private speculation.

3.2 Calculating the Emperor's Death

The political danger of predicting the lifespan, downfall, or date of death of a reigning emperor.

3.3 Maiestas and the Expansion of Treason

How Roman law connected divination, conspiracy, succession, and offenses against imperial authority.

3.4 Predictions That Could Produce Their Own Fulfillment

How a predicted date might coordinate military officers, senators, heirs, rivals, or assassins.

3.5 Astrologers, Informers, and Imperial Paranoia

The use of prosecutions, expulsions, surveillance, and exemplary punishment to control predictive knowledge.

3.6 The Date as an Instrument of Conspiracy

Why the danger lay not in chronology itself but in the capacity of a date to organize political action.

4. Medieval Numerals, the Arrival of Zero, and the Regulation of Commercial Notation

4.1 Roman Numerals and the Pre-Positional Commercial World

The strengths and limitations of traditional European systems of calculation and recordkeeping.

4.2 The Transmission of Hindu-Arabic Numerals

The movement of positional notation through India, the Islamic world, Mediterranean trade, and European mathematics.

4.3 Fibonacci and the Practical Power of the New Numerals

The use of positional notation in commerce, currency conversion, interest calculations, weights, and measures.

4.4 Florence and the 1299 Restriction

The commercial regulation often simplified into the claim that Florence “banned zero.”

4.5 Zero as a Suspected Instrument of Fraud

Fears that circles, appended zeros, and easily modified digits could alter the value of handwritten financial records.

4.6 Numerical Regulation as Data-Integrity Policy

The Florentine restriction as an early attempt to control authentication, legibility, and documentary trust.

4.7 From Forbidden Symbol to Foundation of Computation

The eventual transformation of zero from a suspicious foreign notation into a central element of modern mathematics and digital technology.

5. From Telegraph Codes to Telephone Phreaking: Numbers Begin Controlling Machines

5.1 The Mechanization of Symbolic Communication

How telegraphy, switching systems, signaling protocols, and automated networks gave codes direct operational effects.

5.2 Control Information Inside the Communication Channel

The vulnerability created when commands and ordinary user communications traveled through the same system.

5.3 The 2600-Hz Telephone Signal

The role of a specific frequency in long-distance telephone-network supervision.

5.4 Whistles, Blue Boxes, and Unauthorized Routing

How users reproduced internal signaling tones to manipulate telephone switches and avoid billing.

5.5 When a Frequency Becomes a Lockpick

Why 2600 Hz was lawful to study or generate in isolation but potentially criminal when intentionally transmitted into a telephone network.

5.6 The Birth of Machine-Interpreted Illegality

The transition from numbers that symbolized prohibited ideas to signals that caused infrastructure to perform unauthorized actions.

6. Cryptography, Export Controls, and the Recognition of Source Code as Speech

6.1 Cryptography as a Military and National-Security Technology

The historical treatment of strong encryption as a strategic capability rather than ordinary consumer software.

6.2 Algorithms, Implementations, and Machine Readability

The legal distinction between describing an encryption method and distributing an executable implementation.

6.3 The Karn Controversy

The contrasting treatment of cryptographic source code printed in a book and the same material stored on a disk.

6.4 Bernstein and the First Amendment

The legal argument that computer source code communicates scientific ideas and therefore possesses expressive value.

6.5 Source Code as Speech and Machine

The dual nature of code as an explanation for humans and an instruction for computers.

6.6 The Collapse of the Print-Digital Boundary

How scanners, optical character recognition, automated transcription, and AI increasingly erase the distinction between printed and executable information.

6.7 Functional Speech as a New Legal Category

The unresolved constitutional problem of information that communicates and acts at the same time.

7. Copyright Circumvention and the Golden Age of the Illegal Number

7.1 The Digital Millennium Copyright Act

The emergence of legal prohibitions against circumventing technological protection measures and trafficking in circumvention tools.

7.2 DeCSS and DVD Encryption

The development and distribution of software capable of descrambling protected DVD content.

7.3 The 1,401-Digit Illegal Prime

The encoding of DeCSS-related software into a mathematically notable prime number.

7.4 Representation Invariance

Why binary, hexadecimal, decimal, compressed, visual, musical, and textual forms may contain the same underlying information.

7.5 The AACS HD DVD and Blu-ray Processing Key

The 2007 disclosure of a hexadecimal key and the campaign to remove it from websites.

7.6 The Internet Revolt and the Streisand Effect

The mass republication of the key in articles, images, songs, flags, clothing, and other forms.

7.7 The Free Speech Flag

The encoding of key bytes as colors and the resulting challenge to literal content prohibition.

7.8 The PlayStation 3 Root-Key Dispute

Sony’s litigation over console security, circumvention tools, technical publication, and unauthorized access.

7.9 The Windows XP FCKGW Product Key

A leaked volume-license credential as a weaker but revealing example of a string functioning as commercial authorization.

7.10 Why No Integer Is Illegal in Every Context

The distinction between publishing a mathematical sequence, trafficking in a circumvention device, and using a key to obtain unauthorized access.

8. Political Numbers, Historical Memory, Extremist Codes, and Alleged Threats

8.1 Numbers as Compressed Political Language

Why short numerical expressions can evade censorship, identify communities, preserve memory, or communicate hostility.

8.2 The Meaning of 8964

The use of a four-digit date code to commemorate the June 4, 1989, Tiananmen Square crackdown.

8.3 Censorship, Search Filtering, and Platform Enforcement

The suppression of numerical references through content moderation, product restrictions, surveillance, and public-order enforcement.

8.4 The Meaning of 1312

The conversion of ACAB into alphabet-position notation and its use in anti-police protest.

8.5 Contextual Enforcement in Europe

The legal importance of whether a slogan attacks an institution generally or identifiable individual officers.

8.6 The Meaning of 1488

The combination of white-supremacist numerical symbols and their recognition by extremist movements.

8.7 License Plates and Administrative Symbol Control

Government refusal to issue particular combinations without prohibiting their appearance in scholarship, journalism, or arithmetic.

8.8 The Meaning of 187

The migration of California's murder-statute number into music, gang language, popular culture, and threatening communications.

8.9 The 8647 Controversy

The competing interpretations of a numerical slang expression associated with the 47th president.

8.10 The James Comey Prosecution

The distinction between an indictment, a disputed interpretation, proof of threatening intent, and a final judicial finding.

8.11 True Threats and Mental State

Why criminal liability requires attention to the speaker, target, context, foreseeable interpretation, and constitutionally required level of intent.

8.12 The Problem of Ambiguous Numerical Speech

How slang, satire, political opposition, historical reference, coded incitement, and genuine threats can share the same visible form.

9. Credentials, Account Numbers, Passwords, and Numbers That Confer Authority

9.1 Access Devices in Modern Law

The legal treatment of account numbers, cards, PINs, passwords, electronic serial numbers, and authentication codes.

9.2 Descriptive Strings and Capability Strings

The distinction between a number that identifies something and a number that authorizes an action.

9.3 Bearer Capabilities

Why practical control of a password, private key, reset token, or signing credential may be equivalent to control of an account or asset.

9.4 Authorized Possession and Criminal Possession

How identical digits can be lawful in the hands of an account holder and unlawful when stolen or trafficked for fraud.

9.5 The Changing Status of a Credential

The progression from valid and authorized to leaked, revoked, expired, blacklisted, or evidentiary.

9.6 Hash Values as Numerical Fingerprints

The use of cryptographic hashes to identify malware, stolen files, or known prohibited material.

9.7 A Hash Is Not the Underlying File

Why an identifier associated with contraband is not ordinarily equivalent to possession of the contraband itself.

9.8 False Matches, Secret Lists, and Procedural Fairness

The risks created when governments or platforms rely on opaque numerical matching systems.

10. Cryptocurrency Addresses, Sanctions, and Autonomous Code

10.1 Public Addresses and Private Keys

The difference between identifying a blockchain destination and possessing the authority to move its assets.

10.2 Cryptocurrency Addresses in Sanctions Records

The use of alphanumeric wallet identifiers to trace or restrict dealings with sanctioned actors.

10.3 Publishing an Address Versus Transacting With It

Why journalism, research, compliance analysis, and blockchain investigation differ from transferring blocked property.

10.4 Tornado Cash and the Limits of Property-Based Sanctions

The legal dispute over whether immutable smart contracts can be treated as sanctionable property.

10.5 Code Without an Owner

The emergence of autonomous software that continues to operate after its creators lose the ability to modify or terminate it.

10.6 Smart-Contract Addresses as Entrances to Processes

How an address can identify not merely an account but a persistent, executable mechanism.

10.7 Decentralization and the Failure of Traditional Seizure

Why a distributed program may remain available even after websites, companies, or developers are sanctioned or prosecuted.

10.8 The Legal Problem of Unstoppable Execution

The challenge of regulating processes that no single person, corporation, server, or jurisdiction controls.

11. Exploit Strings, Adversarial AI Tokens, Synthetic DNA, and Information Hazards

11.1 Malicious Inputs as Operational Strings

How database injections, malformed files, network packets, and parser exploits convert ordinary-looking characters into commands.

11.2 Authorization as the Legal Boundary

Why identical exploit strings may be legitimate in an approved security test and criminal when used against an unauthorized target.

11.3 Adversarial AI Suffixes

The optimization of unusual token sequences designed to bypass refusals or manipulate machine-learning systems.

11.4 Machine Meaning Without Human Meaning

Why a string that appears nonsensical to a person may have a precise effect within a model's statistical geometry.

11.5 Model Dependence and Ephemeral Functionality

The dependence of jailbreak strings on model architecture, tokenizer, prompt context, system instructions, and software version.

11.6 DNA as Executable Biological Information

The representation of biological instructions through strings composed of A, C, G, and T.

11.7 Sequence Screening and Synthetic Biology Governance

The attempt to detect dangerous orders without obstructing legitimate biomedical and scientific research.

11.8 Fragments, Assembly, and the Missing Decoder

Why individually harmless components can become dangerous when combined with other sequences, tools, organisms, or laboratory procedures.

11.9 Algorithms That Could Collapse Cryptographic Security

The information-hazard implications of methods capable of defeating widely used encryption or authentication systems.

11.10 Distributed Knowledge and Threshold Possession

The legal difficulty of assigning responsibility when no single participant possesses the complete dangerous sequence or method.

11.11 The Convergence of Code and Matter

The transition from strings that influence machines to strings that can help construct physical biological systems.

12. A General Theory of Illegal Numbers and the Future of Executable Information

12.1 Numbers Are Rarely Illegal as Mathematical Objects

The central distinction between an abstract value and conduct involving one of its representations.

12.2 A Taxonomy of Restricted Numbers

Secret numbers, restricted numbers, unauthorized credentials, prohibited-use strings, symbolic codes, evidentiary strings, and contraband-encoding numbers.

12.3 The Equivalence Class of Representations

The principle that one informational object may have unlimited decimal, binary, visual, auditory, linguistic, encrypted, or compressed forms.

12.4 Why Universal Numerical Blacklists Cannot Work

The ease of re-encoding, fragmenting, encrypting, disguising, or embedding restricted information.

12.5 The Theoretical Limits of Automated Classification

Why no perfect system can determine every possible consequence of every possible program or input.

12.6 False Positives and False Negatives

The inevitable errors produced by numerical blacklists, pattern matching, behavioral detection, and symbolic interpretation.

12.7 The Essential Variables of Legal Meaning

String, representation, function, target, authorization, intent, audience, context, and consequence.

12.8 Regulating Conduct Rather Than Integers

The case for focusing on unauthorized access, trafficking, targeted threats, harmful transactions, and demonstrable assistance.

12.9 Protecting Research, Journalism, Mathematics, and Historical Memory

The need for safeguards covering legitimate analysis, documentation, criticism, security research, and scholarship.

12.10 From Symbolic Meaning to Causal Power

The historical progression from sacred taboos and political dates to commands, credentials, executable contracts, AI exploits, and biological sequences.

12.11 When Speech Becomes Infrastructure

The erosion of the boundary between describing an action and causing a machine to perform it.

12.12 The Future of Numerical Governance

The challenge of constructing legal systems for information that can authorize, execute, reproduce, manufacture, persuade, and persist autonomously.

13. References

Art

1. Sacred Numbers, Forbidden Knowledge, and the Ancient Origins of Numerical Control

The history of "illegal numbers" does not begin with computers, cryptographic keys, or modern statutes. It begins with the much older belief that numbers reveal the hidden organization of reality. In many ancient societies, numerical relationships were not regarded as neutral tools for counting. They could express divine order, musical harmony, astronomical cycles, ritual purity, political legitimacy, or the architecture of the cosmos. Knowledge of numbers could therefore become sacred knowledge, restricted knowledge, or politically dangerous knowledge.

The ancient examples differ fundamentally from modern prohibitions against unauthorized computer access or technological circumvention. There was no legislative code declaring that a particular irrational magnitude was unlawful to possess. Nevertheless, the social mechanisms surrounding secret doctrine, initiatory knowledge, religious authority, and communal discipline created an early form of numerical control. A discovery could be dangerous not because the number itself caused a machine to act, but because it contradicted an institution's account of how the universe was ordered.

The Pythagorean encounter with incommensurable magnitudes provides the most famous example. The diagonal of a square revealed that geometry contained relationships that could not be expressed as ratios of whole numbers. Later tradition transformed this discovery into a dramatic story of secrecy, betrayal, and death. Whether or not the story is historically accurate, it captures a recurring pattern that will continue throughout this paper: information becomes restricted when its circulation threatens an established system of authority.

1.1 Numbers as Religious and Cosmological Principles

Modern mathematics generally treats numbers as abstract objects whose truth does not depend on religion, government, or culture. Ancient numerical thought was often less sharply divided from theology, music, astronomy, medicine, and philosophy. Numbers could represent qualities as well as quantities. Particular numerical arrangements might signify balance, justice, completeness, masculinity, femininity, harmony, or divine order.

This way of thinking arose partly from observations that appeared to connect simple numerical ratios with the structure of nature. The recurring phases of the Moon, the periodic motions of celestial bodies, the proportions of geometric figures, and the intervals of music all suggested that order could be expressed numerically. A stretched string divided into simple ratios produced recognizable musical consonances. The octave corresponded to a 2:1 relationship, the fifth to 3:2, and the fourth to 4:3. These relationships encouraged the belief that pleasing sound was not accidental but arose from an underlying arithmetic structure.

Once harmony was understood numerically, it was natural to extend the principle. Perhaps the heavens, the human body, the moral life, and the political community were also governed by

proportion. Number could then become more than a method of measurement. It became an explanatory principle linking separate domains of existence.

In such a worldview, mathematical knowledge carried spiritual significance. Learning a numerical relationship could be understood as discovering part of the cosmic order. Conversely, a mathematical result that did not fit the accepted system could appear to threaten more than a technical theory. It could call into question the assumed unity of mathematics, nature, and the divine.

This fusion of numerical and sacred meaning created conditions in which mathematical information might be guarded. Knowledge could belong to initiates rather than to the general public. A theorem might be transmitted as part of a way of life involving discipline, ritual, loyalty, and moral instruction. Restriction was maintained not by modern copyright or national-security law but by membership, oath, reputation, exclusion, and religious obligation.

1.2 The Pythagorean Worldview

Pythagoras is believed to have lived during the sixth century BCE and to have established a community in the Greek city of Croton in southern Italy. Reconstructing his teachings is difficult because he left no surviving writings, and many accounts were composed centuries later. The historical Pythagoras became surrounded by legend, while doctrines developed by later Pythagoreans were often attributed retrospectively to the founder.

Despite this uncertainty, the tradition associated with Pythagoras clearly connected mathematics with a disciplined philosophical and religious life. The Pythagorean community studied arithmetic, geometry, music, and astronomy while also observing rules governing diet, conduct, memory, initiation, and communal loyalty. Mathematics was not merely vocational training. It was part of the purification and ordering of the soul.

The phrase “all things are number,” although simplified in popular retellings, expresses an important tendency in Pythagorean thought. Whole numbers and their ratios seemed capable of explaining patterns across nature. Musical intervals offered especially persuasive evidence. If consonance could be generated by ratios involving small integers, perhaps beauty and order themselves were grounded in number.

Whole numbers possessed geometric forms as well. Collections of points could be arranged into triangular, square, oblong, and other figurate numbers. Arithmetic and geometry therefore appeared to support each other. A number could be visualized spatially, and a geometric pattern could be analyzed numerically.

The first four positive integers had special importance because $1 + 2 + 3 + 4 = 10$. These numbers could be arranged as a triangular figure known as the tetractys. Later Pythagorean

tradition associated it with harmony and cosmic order. The tetractys illustrates how a numerical form could function simultaneously as mathematics, symbol, and sacred object.

This intellectual system did not necessarily claim that every length must itself be a whole number. Measurement normally involves ratios. A length of three units compared with another of two units gives the ratio 3:2. The deeper expectation was that magnitudes could be made commensurable: that two lengths would share some sufficiently small common unit allowing both to be expressed as whole multiples of it.

The diagonal of a square disrupted that expectation.

1.3 The Irrationality of the Square Root of Two

Consider a square whose side has length 1. By the geometric relation later called the Pythagorean theorem, the square of the diagonal's length equals the sum of the squares of the two sides:

$d^2 = 1^2 + 1^2 = 2$

Therefore:

$d = \sqrt{2}$

The difficulty is that $\sqrt{2}$ cannot be expressed as a ratio of two integers. No fraction p/q, however large its numerator and denominator, is exactly equal to $\sqrt{2}$.

The familiar decimal 1.41421356... is only an approximation. The digits continue without terminating or entering a repeating pattern. Ancient Greek mathematics did not initially frame the issue in terms of infinite decimal expansions. The central concept was incommensurability.

Two magnitudes are commensurable when some common measure fits into each an exact whole number of times. For example, lengths of 2 units and 3 units are commensurable because a unit length measures one twice and the other three times. The side and diagonal of a square are incommensurable because no common length, however small, measures both exactly as whole multiples.

A classical proof can be expressed through parity. Assume that $\sqrt{2} = p/q$, with p and q whole numbers having no common factor. Squaring both sides gives:

$p^2 = 2q^2$

Therefore p^2 is even, which means p is even. Let $p = 2r$. Substitution gives:

$4r^2 = 2q^2$

and therefore:

$q^2 = 2r^2$

Thus q is also even. But if both p and q are even, they share the factor 2, contradicting the original assumption that p/q was in lowest terms. The assumption that √2 is a ratio of integers must therefore be false.

For modern readers, this is an elegant elementary proof. In its ancient intellectual setting, however, the result had profound implications. It showed that a simple geometric object contained a magnitude that escaped exact expression as a whole-number ratio. Geometry was therefore richer than a system built exclusively from commensurable numerical proportions.

It would be misleading to say that the discovery destroyed Greek mathematics. It instead stimulated new mathematical concepts and more rigorous treatments of ratio and magnitude. Yet it did expose a boundary within the earlier numerical worldview. The square, an emblem of order and symmetry, contained within itself a diagonal that could not be captured by the arithmetic relationships expected of it.

1.4 Hippasus and the Legend of the Drowned Mathematician

Later tradition associates the disclosure of incommensurability with Hippasus of Metapontum, a figure connected with early Pythagoreanism. According to the most dramatic version, Hippasus revealed a forbidden mathematical discovery and was drowned at sea as punishment. Popular retellings often state confidently that the Pythagoreans murdered him for proving that √2 is irrational.

The historical evidence does not justify such certainty. Ancient accounts were written long after the supposed event and do not agree about what Hippasus disclosed. Some traditions connect his fate with incommensurable magnitudes. Others associate it with the geometric construction of a regular dodecahedron. Still others present his drowning as divine punishment for revealing secret teachings rather than as an execution carried out by fellow Pythagoreans.

Even Hippasus's precise role in the discovery of irrationality remains uncertain. He may have been connected to an early proof concerning the diagonal of a square, or his name may have become attached to the story because he represented dissent within the Pythagorean tradition. The available evidence does not permit a confident reconstruction.

The legend nevertheless became culturally powerful because it dramatizes a recognizable conflict. A community possesses a system of knowledge. A member discovers or reveals something that threatens the system. The institution responds by suppressing the information and punishing the discloser.

In this form, Hippasus becomes an archetype of the scientific dissident. His story resembles later narratives involving censored astronomers, prosecuted cryptographers, whistleblowers,

hackers, and publishers of restricted information. The specific mechanisms differ, but the narrative structure remains stable: forbidden knowledge escapes from a controlled circle and challenges the authority that attempted to contain it.

The legend should therefore be preserved in this history, but it must be labeled as legend. Its significance lies less in proving that an ancient mathematician was murdered than in showing how later cultures imagined the danger of a mathematical truth that an institution could not assimilate.

1.5 Secrecy Versus Statutory Illegality

Calling √2 an "illegal number" is rhetorically effective but historically imprecise. No surviving Greek statute prohibited citizens from writing, calculating, or discussing the diagonal of a square. There was no ancient equivalent of a modern criminal code defining possession of an irrational magnitude as an offense.

The relevant category is esoteric or restricted knowledge. A philosophical or religious community may reserve teachings for members who have completed prescribed stages of initiation. Disclosure can then be treated as betrayal, sacrilege, or violation of an oath. Enforcement occurs through expulsion, condemnation, loss of status, ritual sanction, or fear of supernatural punishment.

Statutory illegality is different. It depends upon a recognized governing authority, a defined prohibition, and an institutional process for investigation and punishment. Modern cases involving computer access, sanctions, threats, or technological circumvention generally operate within this legal framework.

The distinction matters because the phrase "illegal number" can otherwise collapse very different phenomena. A sacred convention is not identical to a criminal statute. A secret theorem is not identical to a stolen password. A politically censored date is not identical to an encryption key. Each case involves a different relationship among information, authority, intention, and consequence.

Nevertheless, secrecy and legality share an underlying logic. Both attempt to regulate not the mathematical object itself but access to one of its meaningful representations. The value √2 cannot be eliminated from geometry. What can be controlled is who may know a proof, who may communicate it, and what consequences follow from disclosure.

1.6 Numerical Knowledge as a Threat to Authority

The Pythagorean example introduces the paper's central theme: information becomes dangerous when it crosses from abstract knowledge into institutional consequence.

An unexpected mathematical result can threaten authority in several ways. It may contradict a sacred doctrine, expose the limits of an accepted theory, weaken the prestige of teachers, or provide outsiders with knowledge reserved for an elite. Suppression may therefore be motivated by fear of social disorder as much as by concern for intellectual consistency.

The irrationality of √2 also demonstrates the futility of attempting to prohibit a mathematical truth. Once the relationship has been proved, it does not depend on the continued existence of the person who discovered it. Every unit square reproduces the problem. The knowledge can be independently rediscovered by anyone capable of following the geometry.

This property distinguishes mathematical information from ordinary physical contraband. A physical object can be confiscated or destroyed. An abstract relationship cannot. Its representations can be burned, hidden, censored, or restricted, but the underlying truth remains available for reconstruction.

The same principle will reappear in later periods. A cryptographic algorithm can be reimplemented from its description. A software program can be encoded as a prime number. A prohibited key can be rendered as colors or music. A censored historical date can be compressed into four digits. A blockchain program can persist after its creators lose control of it. A biological sequence can be copied from a digital record.

The history of illegal numbers therefore begins with a paradox. Authority may control teachers, manuscripts, institutions, and channels of communication, but it cannot legislate a mathematical relationship out of existence. The struggle is always over access, representation, interpretation, and use.

In the ancient world, the feared number threatened a cosmology. In the modern world, a number may unlock a machine, transfer property, identify a target, or generate a biological product. The causal power changes, but the governing impulse remains recognizable: when a sequence appears capable of destabilizing an established order, those who depend upon that order attempt to control its circulation.

2. Sacred Writing, Numerical Substitution, and the Hebrew Treatment of Fifteen and Sixteen

The Hebrew treatment of the numbers fifteen and sixteen offers a different kind of numerical restriction from the Pythagorean tradition. Here, the problem was not that the quantities themselves challenged a philosophical system. Fifteen and sixteen were ordinary integers. The difficulty arose because Hebrew letters also functioned as numerals, and the most direct written forms of these two numbers resembled part of the sacred name of God.

The resulting convention demonstrates that a number can become sensitive because of its notation rather than its value. Arithmetic remained unchanged. Fifteen was still fifteen, and sixteen was still sixteen. What changed was the sequence of letters chosen to represent them.

Instead of writing fifteen as ten plus five and sixteen as ten plus six, Hebrew convention uses nine plus six and nine plus seven. The alternative forms preserve the numerical values while avoiding letter combinations that could be read as abbreviations of the divine name.

This is not an example of statutory criminalization. It is a religious and orthographic convention shaped by reverence. Yet it belongs in the history of restricted numbers because it reveals a fundamental principle: numerical symbols do not always remain purely numerical. When an alphabet also serves as a numbering system, a string may simultaneously represent a quantity, a word, a name, and a sacred object.

The Hebrew case therefore introduces the distinction between a number and its representation. That distinction will become central in later sections. A modern cryptographic key may be written in hexadecimal, decimal, binary, colors, or music while retaining the same functional information. In Hebrew numerical notation, the opposite strategy appears: the same numerical value is deliberately expressed through a different symbolic sequence so that an unwanted linguistic meaning is not produced.

2.1 Letters as Numbers

Before the widespread adoption of the modern Hindu-Arabic numeral system, many cultures used letters of the alphabet to represent numbers. Greek, Hebrew, and other ancient writing systems assigned numerical values to individual letters. In such systems, arithmetic notation and ordinary language shared the same visible symbols.

Traditional Hebrew numerical notation assigns values to the letters in ascending order. The first nine letters represent the numbers one through nine. Later letters represent the tens from ten through ninety, and the final letters represent larger values such as one hundred, two hundred, three hundred, and four hundred.

For example, the letter aleph represents one, bet represents two, and gimel represents three. Yod represents ten. Kaf represents twenty. Qof represents one hundred. Larger numbers are written by combining letters whose values add together.

This method differs from the positional system used in modern decimal notation. In the number 15, the first digit does not simply mean one; its position gives it the value ten. Traditional Hebrew notation is additive rather than positional. A sequence representing ten and five has the value fifteen because the individual values are summed.

The system creates a close relationship between text and arithmetic. A letter can belong to a word in one setting and represent a quantity in another. The distinction depends upon context, marks, surrounding text, and convention.

This dual use gives written numbers semantic possibilities that do not normally occur with modern digits. The sequence “15” does not ordinarily resemble an English divine name. A Hebrew letter combination representing fifteen, however, can also resemble a sacred linguistic expression. The problem is therefore not numerical in the abstract. It arises because alphabetic symbols carry several forms of meaning at once.

The Hebrew example illustrates a broader historical truth. Writing systems are not neutral containers. Their structure influences what can be expressed easily, what ambiguities arise, and what social rules develop around them. A society using letters as numbers encounters problems different from those faced by a society using separate numerical symbols.

2.2 The Divine Name and Reverential Avoidance

The Hebrew Bible contains a four-letter divine name conventionally represented in English scholarship as YHWH. In Jewish tradition, this name is treated with exceptional reverence. It is generally not pronounced as written in ordinary religious reading. Substitute titles are used instead.

The written name is also handled with care. Sacred texts containing divine names may be subject to special rules concerning copying, erasure, storage, and disposal. This reverential treatment extends beyond pronunciation to the visual form of the letters themselves.

The ordinary additive representation of fifteen would be ten plus five. In Hebrew letters, ten is represented by yod and five by he. Together, these letters form a sequence resembling the beginning of the divine name.

The ordinary representation of sixteen would be ten plus six. Six is represented by vav. The combination of yod and vav likewise resembles a shortened sacred form.

The problem does not mean that every occurrence of these letters is automatically equivalent to writing the full divine name. Context matters, and Jewish legal and scribal traditions contain detailed distinctions concerning sacred intention, exact spelling, and the status of particular written forms. Nevertheless, customary practice avoids these combinations when writing the numbers fifteen and sixteen.

This avoidance reflects a principle of caution around sacred language. A numerical notation should not casually generate a sequence that may be interpreted as a divine name. Rather than forcing readers to determine repeatedly whether the combination is sacred or merely arithmetic, the writing convention removes the ambiguity.

The practice also prevents ordinary numerical inscriptions from placing sacred letter combinations in unsuitable contexts. Numbers appear on calendars, lists, pages, buildings, receipts, and everyday objects. Alternative notation reduces the likelihood that a revered form will be written where it may later be erased, discarded, damaged, or treated casually.

Here, numerical restriction is motivated not by fear of the number but by respect for the written name. Fifteen and sixteen are not unlucky, dangerous, or forbidden quantities. Their expected alphabetic forms are avoided because the symbols participate in sacred language.

2.3 Nine Plus Six and Nine Plus Seven

The solution is mathematically simple. Fifteen is written as nine plus six:

9 + 6 = 15

Sixteen is written as nine plus seven:

9 + 7 = 16

In Hebrew notation, this replaces the combinations associated with ten plus five and ten plus six. The alternative sequences produce the correct numerical values without resembling the sacred forms.

This convention is important because it shows that representation can be changed without changing the underlying object. The quantity remains invariant while its visible encoding is altered.

The operation can be written abstractly as:

15 = 10 + 5 = 9 + 6

16 = 10 + 6 = 9 + 7

The equality of the expressions is ordinary arithmetic. Their cultural acceptability differs because the symbols used on the right side carry linguistic associations.

This is one of the earliest examples in the paper of deliberate re-encoding. Later, software will be transformed into a large prime number, an encryption key will be converted into colors, and restricted information will be embedded in music or images. In those cases, re-encoding is often used to evade regulation or demonstrate its futility. In Hebrew notation, re-encoding serves the opposite purpose. It avoids producing a sensitive sequence.

Both situations depend on the same distinction:

value or information ≠ particular representation

The Hebrew convention also demonstrates that there is rarely only one way to write a number. Modern readers may assume that fifteen is naturally and uniquely written as “15,” but this is merely one notation. It could also be written as XV, 1111 in binary, F in hexadecimal, fifteen in words, or through numerous additive combinations.

Once multiple representations are available, societies may choose among them for reasons extending beyond mathematical efficiency. They may prefer a form because it is sacred, traditional, legible, secure, politically acceptable, or less susceptible to fraud.

The use of nine plus six and nine plus seven is therefore not a corruption of mathematics. It is an adaptation of notation to religious and linguistic context.

2.4 Numerical Notation as Sacred Language

In alphabetic numbering systems, the boundary between language and mathematics becomes unusually porous. A sequence may be interpreted as a sum of numerical values, as a cluster of letters, or as part of a word. Its meaning depends on the expectations of the reader.

This layered significance is intensified when the letters form or resemble a sacred name. The written form is no longer merely a convenient label for a quantity. It participates in a tradition that assigns holiness, authority, and ritual responsibility to particular words.

Modern legal and technological systems also depend on the difference between human and machine interpretation. A person may see a meaningless hexadecimal string, while a computer recognizes a decryption key. A human may see colored squares, while software reconstructs binary data. The Hebrew case reverses this emphasis. The arithmetic reader sees a number, while the religious reader also perceives a name.

The same symbols therefore operate in two semantic systems:

letters → linguistic and sacred meaning

letters → numerical value

The convention concerning fifteen and sixteen manages the collision between these systems. It prevents a routine numerical operation from inadvertently entering the domain of sacred writing.

This example also warns against assuming that numerical meaning is universal merely because arithmetic is universal. The equality 9 + 6 = 15 does not vary across cultures, but the significance of the letters used to express those quantities can vary profoundly.

A sequence acquires social meaning through language, memory, religion, and institutional practice. The number itself may be abstract, but its written form enters history.

This principle helps explain why later political number codes become powerful. The digits 8964 are mathematically ordinary, yet they can carry the memory of a suppressed historical event. The digits 1312 can serve as an alphabetic substitution. The sequence 1488 can operate as an extremist signal. In each case, numerical notation functions as compressed language.

The Hebrew treatment of fifteen and sixteen is therefore not an isolated curiosity. It is an early and enduring example of numbers becoming words.

2.5 Taboo Without Criminalization

The term “taboo” must be used carefully. In popular language, it may imply fear, irrational prohibition, or punishment. The Hebrew convention is more accurately understood as reverential avoidance.

There is no general claim that the integers fifteen and sixteen are unlawful. There is no attempt to prevent calculation with them, speaking them, counting to them, or using them in measurement. The adjustment concerns how they are written in a particular alphabetic system.

This distinction separates religious convention from statutory prohibition. A criminal law defines conduct that the state may punish through fines, imprisonment, confiscation, or other formal sanctions. A religious convention may be maintained through education, communal expectation, conscience, ritual practice, or interpretation of religious law.

The two can overlap in societies where religious and political authority are joined, but they should not be treated as identical. Calling fifteen an “illegal number” would exaggerate the historical reality. Calling its expected letter combination a prohibited or avoided form is more accurate.

The example nevertheless contributes to a general theory of restricted sequences. It shows that the acceptability of a string can depend on several questions:

What symbols are used?

What else do those symbols mean?

Where are they written?

Who writes them?

With what intention?

How will the text be handled afterward?

These questions will later reappear in secular law. The same encryption key may be lawful in a court filing and unlawful when distributed as part of a circumvention service. A political code

may be protected speech in one setting and evidence of a threat in another. A biological sequence may be legitimate research data or part of an attempt to construct a dangerous organism.

In each case, the string alone does not determine its status. Meaning emerges from context, purpose, and use.

2.6 The Persistence of Ancient Numerical Conventions

The alternative forms for fifteen and sixteen remain visible in modern Hebrew usage. They commonly appear in Hebrew dates, religious calendars, page numbering, chapter references, and other settings where letters serve as numerals.

The convention is especially noticeable in dates. Hebrew calendar years and days may be expressed through letter values, with marks indicating that the sequence is being used numerically. When the day of a month is fifteen or sixteen, the forms based on nine plus six and nine plus seven are used.

This persistence is significant. The practice is not merely an archaeological remnant preserved in manuscripts. It remains part of a living writing system.

Its survival demonstrates how cultural meaning can shape notation for centuries after the original convention has become familiar and automatic. Many users may employ the accepted forms without consciously reconstructing the full theological reasoning each time. The convention becomes embedded in literacy itself.

Such endurance also shows that alternative notation need not impede mathematical understanding. Readers recognize the forms readily. The avoidance of one representation does not erase the quantity or make calculation difficult. A stable substitute solves the symbolic problem.

This differs sharply from attempts to suppress information for which no socially accepted substitute is available. When a government censors a historical date, users invent new codes, images, homophones, or indirect references. When a company tries to remove an encryption key, the key may reappear in transformed formats. Suppression generates adversarial re-encoding.

The Hebrew case is cooperative rather than adversarial. The community shares the reason for avoidance and accepts the alternative notation. The substitution succeeds because writers and readers agree upon its purpose.

This suggests a broader rule. Restrictions on representation are most durable when they are supported by a community’s values and accompanied by a practical alternative. They are least

durable when imposed against unwilling speakers on information that remains easy to transform and redistribute.

The treatment of fifteen and sixteen thus advances the history from secret mathematical knowledge to sacred textual representation. The Pythagorean case concerned a truth allegedly withheld because it threatened a cosmology. The Hebrew case concerns ordinary quantities rewritten to prevent accidental contact with a revered name.

Together, the examples establish two ancient foundations for the history of illegal numbers. First, numerical knowledge may be restricted because of what it reveals. Second, numerical notation may be altered because of what it says.

Later societies will add a third possibility: a number may be controlled because of what it causes a machine to do.

3. Forbidden Predictions, Imperial Death Dates, and Numerical Treason in Rome

Ancient Rome added a political dimension to the control of dangerous numbers. A date could become threatening not because it was mathematically unusual or religiously improper, but because it predicted the death of a ruler. Calculating when an emperor would die could be interpreted as preparation for succession, conspiracy, rebellion, or assassination. The danger lay in the capacity of a forecast to influence conduct.

Roman authorities did not generally prohibit arithmetic, calendars, astronomy, or every form of astrology. Romans routinely consulted omens, interpreted celestial events, observed auspicious dates, and sought information about the future. The state itself maintained religious institutions responsible for interpreting signs believed to reveal divine approval or warning.

The political problem arose when predictive knowledge concerned the emperor's lifespan or the identity of his successor. Such questions touched the most sensitive point in an autocratic system: the transfer of supreme power. An emperor's death date could affect military allegiance, senatorial calculation, dynastic rivalry, financial behavior, and the timing of conspiracies.

The resulting prohibitions were not bans on particular integers. No calendar date was inherently treasonous. A date acquired dangerous significance when attached to the ruler's death and circulated in a politically charged setting. The Roman example therefore advances the history of restricted numbers from sacred representation to actionable political information.

3.1 Astrology as Political Intelligence

Astrology occupied an ambiguous position in Roman society. It could be treated as a learned discipline, a popular service, a foreign superstition, a source of entertainment, or a threat to public order. Its practitioners ranged from respected advisers to itinerant fortune-tellers. Emperors sometimes consulted astrologers privately while restricting their activities publicly.

This apparent contradiction reflected the difference between useful prediction and uncontrolled prediction. An emperor might value information about celestial signs when it was delivered confidentially and interpreted in support of imperial authority. The same practice became dangerous when rivals used it to ask whether the emperor would die soon, whether another man was destined to rule, or whether a planned rebellion would succeed.

Astrological inquiry could therefore function as a form of political intelligence. A forecast about the emperor's future was not merely a private opinion about the stars. It could reveal the intentions of the person asking the question.

A senator who requested his own imperial horoscope might be suspected of ambition. A military commander who asked when the reigning emperor would die might be suspected of preparing to seize power. A group that repeatedly consulted diviners about succession could appear to be testing the timing of a conspiracy.

The astrologer's answer could also influence behavior. A prediction that an emperor would die within a year might encourage hesitant conspirators. A forecast that a rival was destined to rule could attract supporters. A favorable date could be selected for revolt, assassination, or proclamation of a new emperor.

Thus, Roman concern with astrology was partly concern with coordination. The stars themselves did not overthrow rulers. Belief in what the stars foretold could organize people who wanted to do so.

Astrology became especially threatening because it claimed access to information unavailable through ordinary political channels. The emperor controlled official announcements, military commands, legal appointments, and much of public communication. An astrologer appeared to offer knowledge that bypassed this control.

The forecast could therefore compete with imperial authority. The ruler claimed the power to determine succession through adoption, dynastic planning, military command, or legal designation. Astrology implied that succession was already written in the heavens.

3.2 Calculating the Emperor's Death

The most sensitive astrological question concerned the emperor's death. To ask when a private person would die might be treated as curiosity, superstition, or personal anxiety. To ask when the emperor would die carried far greater political implications.

The distinction can be expressed simply:

private death prediction → personal concern

imperial death prediction → possible succession planning

A precise date intensified the danger. General speculation that all rulers eventually die was politically harmless. A forecast identifying a particular year, month, day, or celestial event could become operational. It provided a timetable around which political actors might organize.

Such calculations were especially alarming when combined with questions about who would rule next. A person who asked both when the emperor would die and whether he himself would gain power appeared to be doing more than satisfying philosophical curiosity.

The Roman state consequently treated some forms of divination about the emperor as offenses against public security. The precise legal treatment varied across centuries, rulers, and circumstances. Emperors issued expulsions, imposed restrictions on private consultations, prosecuted individuals for illicit divination, and used accusations of astrological conspiracy within broader treason cases.

The number itself remained ordinary. A year such as 96 CE, for example, was not prohibited as a date. It became politically charged if a prediction associated it with the fall of an emperor.

This relational character is central to the concept of illegal numbers. The legal or political meaning of a sequence may depend on what it is connected to:

date + emperor + predicted death + suspicious intent

Remove the imperial target, and the same date becomes a neutral entry in a calendar. Remove the claim of prediction, and it becomes a historical or administrative notation. Remove the political context, and it loses much of its threatening significance.

Roman restrictions therefore anticipated later true-threat law and modern controls on coded political speech. A sequence does not carry its full meaning alone. Meaning arises from the relationship among speaker, target, audience, purpose, and expected consequence.

3.3 Maiestas and the Expansion of Treason

The Roman concept of maiestas referred broadly to the dignity, sovereignty, or greatness of the Roman people and, increasingly under the emperors, the authority of the ruler. Conduct that diminished this majesty could be prosecuted as treason.

Under the Republic, treason was associated with offenses against the state, military betrayal, and injury to Roman sovereignty. Under imperial rule, the concept expanded as the emperor came to embody the state. An attack upon the emperor could therefore be treated as an attack upon Rome itself.

This expansion created opportunities for broad interpretation. Conspiracy, insulting conduct, disloyal speech, unauthorized consultation about succession, and ominous predictions could be drawn into treason proceedings when authorities believed they threatened imperial security.

Astrological inquiries were rarely judged in isolation. They often appeared alongside other evidence: secret meetings, possession of weapons, suspicious correspondence, political rivalry, sacrifices, magical rituals, or consultations with multiple diviners.

The Roman legal system did not maintain the modern distinction between scientific prediction, religious practice, political speech, and conspiracy with the same clarity expected today. Astrology, magic, sacrifice, and divination could overlap in official perception. An inquiry into the emperor's death might therefore be treated simultaneously as sacrilege, illicit magic, sedition, and preparation for treason.

Maiestas was also vulnerable to political manipulation. Informers could accuse rivals of disloyal speech or prohibited consultation. Emperors could use treason proceedings to eliminate opponents, confiscate property, or intimidate the Senate. The evidentiary value assigned to a prediction might therefore depend as much on imperial politics as on the actual danger it posed.

This makes the Roman case a warning about vague security law. When the offense is defined as diminishing authority rather than performing a clearly specified act, rulers gain wide discretion to reinterpret curiosity, criticism, scholarship, or private belief as political disloyalty.

The forbidden date then becomes not only a possible instrument of conspiracy but also an instrument of prosecution.

3.4 Predictions That Could Produce Their Own Fulfillment

An imperial death prediction was politically dangerous partly because predictions can alter the behavior of those who hear them. A forecast does not merely describe a possible future. It can help create that future.

Suppose an astrologer predicts that an emperor will die during a coming eclipse. Several consequences may follow. Conspirators may choose the eclipse as the time to act. Soldiers may hesitate to defend a ruler believed to be doomed. Rival claimants may prepare their supporters. Officials may transfer wealth, hide documents, or change allegiance. Ordinary citizens may interpret unrelated events as confirmation that imperial authority is collapsing.

The prediction becomes self-reinforcing:

forecast → belief → preparation → political action → apparent fulfillment

This mechanism helps explain why rulers feared even inaccurate predictions. A forecast did not need to arise from valid astronomy or astrology to destabilize the state. It only needed to be believed by people capable of acting upon it.

In a political system without a regular, universally accepted process for succession, uncertainty surrounding an emperor's death was especially dangerous. Roman emperors might be succeeded by sons, adopted heirs, military commanders, senatorial candidates, or victorious usurpers. The death of a ruler could initiate rapid competition among armed factions.

A widely circulated date could reduce uncertainty for conspirators by providing a shared expectation. Each participant might hesitate to act alone, but a prediction could signal that others were preparing for the same moment.

In this way, a date functioned as a coordination device. Its importance did not depend on whether it was objectively true. The date provided a focal point around which expectations could converge.

Modern economics and political theory describe similar phenomena when predictions cause bank runs, market panics, electoral mobilization, or military escalation. Ancient imperial astrology operated through a comparable logic. Belief about the future became a force acting in the present.

3.5 Astrologers, Informers, and Imperial Paranoia

Roman emperors repeatedly confronted the political problem posed by astrologers and diviners. Some rulers expelled astrologers from Rome or Italy. Others prohibited private consultations while retaining favored practitioners at court. Enforcement was inconsistent because emperors themselves often desired predictive knowledge.

This pattern created a monopoly problem. The issue was not always that astrology was considered entirely false or unacceptable. The emperor might instead seek to monopolize its politically useful applications.

Court astrologers could advise the ruler about auspicious dates, rivals, military campaigns, or personal danger. Independent astrologers offered similar services to people outside imperial control. The same knowledge was treated differently depending on who possessed it and whom it served.

Informers played a major role in enforcing political conformity. An accusation that a rival had consulted an astrologer could become valuable evidence in a treason proceeding. Because consultations were often private, cases depended on servants, intermediaries, accomplices, intercepted writings, or testimony obtained under pressure.

This encouraged suspicion. A person might be accused not because a prohibited forecast had actually been produced, but because the alleged consultation fitted a broader narrative of ambition or disloyalty.

Imperial paranoia was not always irrational. Roman emperors faced repeated assassination plots, military revolts, dynastic struggles, and usurpations. Several died violently. In such an environment, inquiries about the emperor's death could reasonably attract attention.

Yet genuine danger and political repression easily became entangled. A ruler who feared assassination might treat all predictive discussion as preparation for murder. An emperor who feared rivals might prosecute their horoscopes rather than their actions. The boundary between prevention and persecution became difficult to maintain.

The political significance of astrology therefore depended on asymmetry. The emperor could ask what the stars foretold about others. Others could not safely ask what the stars foretold about him.

3.6 The Date as an Instrument of Conspiracy

A date can perform several political functions. It can predict, commemorate, threaten, coordinate, authenticate, or command.

In the Roman imperial setting, the forbidden death date combined prediction with coordination. It suggested that the current order had a measurable endpoint and that a transfer of power could be anticipated.

The date could circulate in spoken prophecy, written horoscope, coded letter, ritual consultation, or rumor. Each form provided a means of transmitting political expectation without issuing an explicit command.

A statement such as "the emperor will die on this day" might operate as more than a forecast. Depending on context, it could also imply:

prepare for succession on this day

withdraw support before this day

attack on this day

proclaim the successor after this day

The ambiguity made such statements useful to conspirators and troubling to authorities. The speaker could claim merely to have interpreted the heavens, while listeners could infer a plan.

This ambiguity resembles later coded political expressions. A number may preserve plausible deniability while still communicating meaning to an informed audience. Those outside the group see a date. Those inside it may see an instruction.

The Roman example also illustrates the difference between content and function. The content of the forecast is temporal: it names a point in time. Its function may be political: it organizes expectations and conduct.

The date becomes dangerous through its association with a target and a community of interpreters. A bare sequence such as the third day before the Kalends has no treasonous meaning by itself. Attached to an emperor's predicted death, it becomes potential evidence of conspiracy.

This principle can be stated more generally:

numerical meaning = sequence + social interpretation

political danger = numerical meaning + capacity for coordinated action

Roman authorities attempted to control both stages. They restricted the production of dangerous predictions and punished those who appeared to use them politically.

The policy could never eliminate speculation about imperial death. Every aging or unpopular emperor generated rumors. Every unusual comet, eclipse, illness, or military defeat invited interpretation. Suppression could even increase the prestige of a prediction by suggesting that the ruler feared it.

The result was an ancient version of the problem later called the Streisand effect. Efforts to suppress a forecast could make it more memorable. Prosecution transformed private speculation into public drama. The state's reaction supplied evidence that the prediction mattered.

The Roman history of forbidden death dates therefore reveals an enduring conflict between authority and predictive information. A ruler can punish astrologers, seize manuscripts, exile practitioners, and prosecute suspected conspirators. He cannot eliminate uncertainty about succession or prevent people from assigning meaning to time.

In the Pythagorean tradition, dangerous numerical knowledge threatened a cosmology. In Hebrew notation, a numerical representation risked reproducing a sacred name. In Rome, a date threatened the political order by giving the emperor's power an imagined endpoint.

The progression is significant. Numbers were no longer restricted only because of what they revealed or represented. They could be restricted because of what people might organize themselves to do.

4. Medieval Numerals, the Arrival of Zero, and the Regulation of Commercial Notation

The medieval European encounter with Hindu-Arabic numerals marked a major transition in the history of restricted numbers. Earlier examples involved sacred meaning, philosophical secrecy, or political prediction. The controversy surrounding the new numerals concerned something more practical: trust.

Merchants, money changers, notaries, and civic authorities depended on written records that could establish debts, prices, taxes, contracts, and ownership. A numerical notation was therefore not merely a mathematical convenience. It was part of the legal infrastructure of commerce.

The Hindu-Arabic system offered enormous advantages. Its positional structure made calculation faster, its symbols were compact, and the use of zero allowed the same ten digits to represent numbers of any size. Yet these strengths also created suspicion. The new figures were unfamiliar, easily altered, and capable of changing value through the addition or modification of a single mark.

The best-known example is the Florentine restriction of 1299, often described inaccurately as a ban on zero. The rule was broader and more limited at the same time. It did not abolish zero from mathematics or outlaw Hindu-Arabic numerals everywhere. It restricted their use in certain commercial records, where Roman numerals or written words were considered more resistant to fraud.

The episode shows that numerical systems can become legally sensitive because of their vulnerability to manipulation. The danger was not the abstract value zero. The danger was that notation had become powerful enough to alter financial reality.

4.1 Roman Numerals and the Pre-Positional Commercial World

Before Hindu-Arabic numerals became widespread in Europe, Roman numerals remained deeply embedded in administration, trade, inscriptions, accounting, and legal records. Their symbols represented fixed values:

I = 1

V = 5

X = 10

L = 50

C = 100

D = 500

M = 1,000

Roman numerals were not positional in the modern decimal sense. The value of a symbol did not depend primarily on whether it appeared in the units, tens, or hundreds place. Instead, numbers were formed through combinations of symbols according to additive and, in some cases, subtractive conventions.

For example:

VIII = 8

XII = 12

XXV = 25

Roman numerals had several advantages in documentary settings. A written sequence such as XXVIII was visibly extended. It was not easy to change its value merely by appending a small circular mark. Words could also be written out in full, further reducing ambiguity.

The system was nevertheless cumbersome for calculation. Multiplication, division, extraction of roots, interest calculations, currency conversions, and complex bookkeeping were difficult to perform directly through Roman notation. Practical calculation often relied on counting boards, counters, tables, or the abacus rather than on the written numerals themselves.

This separation between notation and calculation was important. A merchant might calculate with counters and then record the result in Roman numerals or words. The written document served as evidence, while the computational process took place elsewhere.

The Hindu-Arabic system began to collapse that division. The same symbols could be used both to calculate and to record. This increased efficiency but also increased the power of the written mark.

A positional numeral could function as a compact computational object. Its value depended on the exact location of each digit. A small alteration could therefore produce a large change.

4.2 The Transmission of Hindu-Arabic Numerals

The numeral system now used throughout much of the world developed over centuries through contributions from Indian mathematicians and scholars working in the Islamic world. European adoption occurred through trade, translation, education, and contact across the Mediterranean.

The term “Arabic numerals” reflects the route through which many Europeans encountered the system, not its exclusive origin. Indian mathematical traditions had developed positional decimal notation and a symbol for zero. Scholars writing in Arabic studied, extended, and transmitted these methods.

Commercial centers in Italy were especially well positioned to receive the new techniques. Merchants traded with North Africa, the eastern Mediterranean, and the Islamic world. They needed methods for calculating exchange rates, weights, interest, profit, loss, partnerships, and currency conversions.

The numerals entered Europe gradually rather than through a single event. They appeared in manuscripts, astronomical tables, mathematical treatises, and commercial schools. Their adoption varied by region and profession.

This gradual spread created a period in which two numerical cultures coexisted. Traditional notaries and officials often preferred Roman numerals or written words. Merchants and calculation specialists increasingly recognized the efficiency of the new figures.

The disagreement was therefore not simply between knowledge and ignorance. Different groups evaluated the numerals according to different needs. A mathematician valued computational power. A notary valued documentary stability. A merchant might value both but fear fraud.

The controversy illustrates a recurring feature of technological transition. A new system may be objectively more efficient while also creating unfamiliar forms of risk.

4.3 Fibonacci and the Practical Power of the New Numerals

Leonardo of Pisa, commonly known as Fibonacci, played an important role in introducing Hindu-Arabic arithmetic to Latin European readers. His Liber Abaci, first composed in the early thirteenth century, demonstrated the practical advantages of the system.

The work did not present numerals as abstract curiosities. It applied them to commercial problems involving currency exchange, profit, barter, partnerships, measurement, interest, and the valuation of goods.

The title Liber Abaci is sometimes misleading to modern readers because the book was not primarily a manual for the physical abacus. It promoted written calculation using the nine digits and the symbol for zero.

The decimal positional system depends on place value. In the number 507:

5 represents five hundreds

0 represents no tens

7 represents seven units

The zero is essential. Without it, the distinction between 507 and 57 would disappear.

This capacity made large numbers compact and calculations systematic. The same algorithms could be applied to many problems without requiring a separate symbol for every value.

Merchants quickly recognized the advantages. A trader dealing with multiple currencies and fractional units needed more than simple counting. Positional notation allowed computations to be performed on paper, checked, repeated, and taught.

Commercial arithmetic schools emerged in Italian cities to train young merchants. Students learned practical calculation rather than purely theoretical mathematics. The numerals became tools of economic mobility.

Yet the same compactness that made the system powerful made it vulnerable. A value could change through a small alteration. A digit could be modified into another digit. A zero could be appended. A blank space could be filled later.

The new numerals therefore increased both efficiency and the importance of document security.

4.4 Florence and the 1299 Restriction

In 1299, the Florentine guild associated with bankers and money changers issued a regulation restricting the use of the new numerals in financial records. The order is often summarized as a ban on Arabic numerals or even as a ban on zero.

That summary is too broad. The restriction concerned particular commercial contexts. It did not prevent mathematicians from studying the system, merchants from learning it, or individuals from using it privately.

The rule required financial entries to be written in a form considered less vulnerable to alteration, such as Roman numerals or words. The purpose was documentary security rather than philosophical opposition to zero.

This distinction matters. Florence did not declare that zero was false, immoral, or mathematically dangerous. The concern was that written figures could be changed after a transaction had been recorded.

A financial record might determine who owed money, how much interest had accrued, or whether a payment had been made. If the notation could be altered without detection, the legal reliability of the document was weakened.

The restriction can therefore be understood as an early form of compliance policy. Authorities were attempting to standardize how financially significant information was recorded.

The rule also reveals institutional conservatism. New technologies often face resistance from systems built around older methods. Notaries, judges, guild officials, and merchants had established conventions for interpreting Roman numerals and written amounts. The new figures lacked the same history of trusted use.

The difficulty was not merely that the numerals were easy to alter. Officials also needed to read them consistently. A symbol familiar to one merchant might be unfamiliar or ambiguous to another.

Standardization had not yet fully developed. Shapes varied, handwriting varied, and the distinction between certain digits could be unclear. A regulatory authority might reasonably prefer a slower but more familiar system.

4.5 Zero as a Suspected Instrument of Fraud

Zero attracted special suspicion because of its role in positional notation. By itself, zero represented an absence of units. Yet its placement could multiply the value of neighboring digits.

For example:

5 → 50

50 → 500

500 → 5,000

Appending a single zero multiplied the number by ten. Appending two zeros multiplied it by one hundred.

In a handwritten ledger, this created obvious opportunities for fraud. A dishonest person might add a zero to an amount or insert one into a blank space. A small physical mark could produce a large financial change.

Other digits could also be modified. A 1 might be changed into a 7. A 3 might be altered into an 8. Variations in handwriting made detection difficult.

The problem was intensified by the lack of modern safeguards. Medieval records did not benefit from digital timestamps, database logs, photocopies, automated reconciliation, or standardized machine printing. Once a document had been altered, proving the original value could be difficult.

Written words were considered safer because changing “ten” into “one hundred” required more extensive alteration. Roman numerals could also be structured with lines, abbreviations, or surrounding marks that made additions more visible.

Zero therefore became associated with fraud not because it was inherently deceptive, but because it amplified the consequences of a tiny change.

This is a crucial step in the history of illegal numbers. Earlier restrictions concerned what a number meant. The Florentine concern was what a written symbol allowed someone to do.

The zero was becoming operational.

4.6 Numerical Regulation as Data-Integrity Policy

The Florentine restriction can be interpreted as an early data-integrity rule. Its purpose was to preserve the accuracy, authenticity, and legal reliability of financial records.

Modern information security often identifies several basic goals:

confidentiality

integrity

availability

The Florentine problem concerned integrity. Could a record be trusted to remain unchanged after it was written?

A numeral system was acceptable not only if it represented values correctly, but also if unauthorized alterations could be detected or prevented.

This approach anticipates modern practices such as writing amounts in both digits and words on checks. The duplication creates a comparison between two representations. If the figures and words disagree, the document can be questioned.

Similar principles appear in checksums, digital signatures, audit logs, and cryptographic hashes. Each provides a way to detect whether information has been changed.

The medieval solution was comparatively simple: use a notation thought to be harder to modify. Modern systems use mathematical mechanisms to authenticate records.

The underlying concern remains the same:

How can a recipient know that the number now visible is the number originally authorized?

This question is central to banking, contracts, taxation, voting systems, scientific data, and digital communications.

The Florentine episode therefore belongs not only to the history of mathematics but also to the history of cybersecurity. It represents an early recognition that numerical notation can be attacked.

4.7 From Forbidden Symbol to Foundation of Computation

Despite resistance, Hindu-Arabic numerals gradually became dominant in European commerce, science, and administration. Their advantages were too great to ignore.

The positional decimal system simplified arithmetic, supported advanced mathematics, and made written calculation more accessible. Zero became indispensable to algebra, astronomy, engineering, finance, and later computer science.

The same symbol once feared as a vehicle for fraud became a foundation of modern information technology. Binary computation depends on the distinction between zero and one. Digital storage, machine logic, and programming all rely on zero as both value and state.

The historical irony is striking. A symbol restricted in medieval financial records eventually became essential to systems that now secure those records.

Modern banking protects numerical integrity through passwords, encryption, digital signatures, database permissions, and redundant verification. These systems would be impossible without zero and positional notation.

Yet the Florentine concern did not disappear. It changed form. A fraudulent alteration no longer requires ink added to a ledger. It may involve modifying a database entry, intercepting a transaction, changing an account number, or exploiting software.

The lesson was not that zero was dangerous. The lesson was that every efficient system creates new attack surfaces.

A notation powerful enough to transform commerce was also powerful enough to transform fraud. The correct response was not permanent prohibition but the development of stronger methods of authentication and verification.

This pattern will recur throughout the paper. Telephone networks will become vulnerable because signaling and communication share the same channel. Cryptographic systems will be vulnerable because short keys control valuable functions. Software will be vulnerable because input can be interpreted as command. AI systems will be vulnerable because unusual strings can manipulate statistical behavior.

Each new information technology produces both capability and misuse.

The medieval history of zero therefore marks a major transition. Numerical symbols were no longer regulated only because they offended sacred order or threatened political authority. They were regulated because they had become instruments of commercial action.

The number on the page could create debt, transfer value, establish ownership, or commit fraud.

From this point onward, the history of illegal numbers increasingly becomes the history of information systems.

5. From Telegraph Codes to Telephone Phreaking: Numbers Begin Controlling Machines

The transition from medieval notation to modern telecommunications transformed the role of numerical information. Earlier societies regulated numbers because they carried sacred meaning, political implication, or commercial value. With telegraphy and telephony, numbers and coded signals began to operate machinery directly.

A signal no longer merely described an action. It could cause an action.

Electrical communications systems required standardized codes that machines and trained operators could interpret. Pulses, tones, timing intervals, channel numbers, and routing instructions became part of the hidden infrastructure of modern society. These signals controlled where messages traveled, which circuits opened, how calls were billed, and when machines began or ended particular operations.

This created a new kind of vulnerability. A person who understood the control language of a network could imitate its internal commands. An ordinary user might therefore produce a tone or sequence that the system mistook for an instruction from trusted equipment.

Telephone phreaking, especially the exploitation of the 2600-Hz signaling tone, became the defining example. The frequency itself was not illegal. It appeared naturally in sound and could be generated for research, music, or testing. Its legal and technical significance emerged only when it was transmitted into a telephone network for the purpose of manipulating switching equipment or avoiding payment.

The history of 2600 Hz marks a decisive stage in the development of the illegal number. A numerical value had become the specification for a machine-readable key.

5.1 The Mechanization of Symbolic Communication

The telegraph introduced a new relationship between symbols and machines. Written language could be converted into electrical pulses, transmitted across long distances, and reconstructed at the receiving end.

Morse code represented letters and numbers through combinations of short and long signals. The sequence was meaningful to trained operators, but it also belonged to an increasingly automated system of relays, sounders, printers, and switching equipment.

This was more than faster writing. Communication had become encoded into physical events with precise timing and electrical characteristics.

As networks expanded, operators needed additional codes for routing, billing, service messages, priority handling, and equipment control. Some codes represented human language, while others instructed the network itself.

The distinction can be expressed as follows:

message code → information intended for a person

control code → information intended for the system

The two kinds of information often traveled through the same wires. That design was efficient, but it created a fundamental security problem. If users could reproduce control signals, the network might be unable to distinguish an authorized instruction from an imitation.

This problem would later recur in computer systems. A database might interpret user text as a command. A web application might confuse data with executable code. An AI system might interpret an adversarial suffix as a privileged instruction.

Telephone phreaking was therefore not an isolated historical curiosity. It was an early instance of a general failure to separate user content from system control.

5.2 Control Information Inside the Communication Channel

Early long-distance telephone systems used signaling tones to manage connections between switching centers. The network needed to know when a trunk line was occupied, when it had been released, which destination was requested, and how the call should be routed.

In many systems, these control signals traveled in-band. This meant that they used the same audible channel that carried the caller’s voice.

In-band signaling offered practical advantages. Existing voice circuits could carry both conversation and network instructions without requiring a completely separate control network.

The weakness was equally important. Because the control frequencies existed inside the audible range, a subscriber could potentially hear, record, reproduce, or imitate them.

The network assumed that certain tones originated only from trusted switching equipment. That assumption was false.

Once technically curious users identified the relevant frequencies and timing patterns, they could experiment with network behavior. They discovered that the system responded to accurately generated tones regardless of whether those tones came from official equipment or from a device connected to an ordinary telephone.

The machine authenticated the signal by form rather than origin.

This is a primitive version of a problem that remains central to cybersecurity. A system may verify that an input has the correct structure without verifying that the sender has the authority to issue it.

The telephone network asked, in effect:

Does this sound like a valid command?

It did not always ask:

Did this command come from an authorized source?

5.3 The 2600-Hz Telephone Signal

A frequency of approximately 2600 Hz played a critical role in certain long-distance telephone systems. When transmitted continuously on a trunk line, it indicated that the line was idle or available.

When a call was established, the tone disappeared. When the connection ended and the trunk returned to an unused state, the tone reappeared.

This signaling arrangement allowed switching equipment at different locations to coordinate the status of long-distance circuits. The tone was not intended for subscribers. It belonged to the network's internal control language.

Phreakers discovered that transmitting a 2600-Hz tone during a call could cause remote equipment to interpret the trunk as having been released, even though the caller's local connection remained active.

This produced an unusual state. The subscriber was still connected to the long-distance trunk, but the remote system believed the previous call had ended. Additional routing signals could then be transmitted to direct the trunk toward another destination.

The frequency functioned like a reset command. It did not contain a complete destination or billing instruction by itself, but it placed the network into a state from which further manipulation became possible.

The number 2600 therefore acquired a dual identity:

2600 as a frequency measurement

2600 as a network-control capability

The distinction depended entirely on context. In an acoustics laboratory, 2600 Hz was an ordinary tone. Inside a vulnerable telephone trunk, it could change the state of national infrastructure.

5.4 Whistles, Blue Boxes, and Unauthorized Routing

One of the most famous stories in phreaking history concerned toy whistles distributed in cereal boxes. Some whistles reportedly produced a tone close to 2600 Hz when one of their holes was covered.

The whistle itself was simple, but the network response was sophisticated. A user who produced the correct frequency could interfere with trunk supervision.

More advanced users built electronic devices known as blue boxes. These generated 2600 Hz and additional multifrequency signaling combinations used by telephone switching systems.

The blue box allowed a user to seize or reset a trunk and then transmit routing commands resembling those sent by official operators. With sufficient knowledge, a phreaker could place long-distance or international calls without the normal billing process.

The device did not physically break into a telephone exchange. It impersonated the exchange's signaling language.

This is an important conceptual distinction. Traditional burglary overcomes a physical barrier. Phreaking overcame a protocol.

The blue box converted technical knowledge into unauthorized capability:

knowledge of frequency + signal generator + network access = control

The device therefore resembled later digital tools. A software exploit, stolen password, cryptographic key, or adversarial token sequence may also grant access without physically damaging the target.

Telephone companies and law-enforcement agencies treated blue-box use as theft of service, fraud, or unauthorized interference with communications systems. The prohibited conduct involved deliberate manipulation and avoidance of payment, not mere knowledge of the frequencies.

A person could lawfully study telephone signaling, build test equipment, or discuss the operation of the network. Legal risk increased when the equipment was used to obtain services without authorization.

The distinction between research and misuse would become a recurring legal problem in computer security.

5.5 When a Frequency Becomes a Lockpick

A frequency is a physical property of a repeating wave. It is measured in cycles per second, or hertz. No government can plausibly prohibit the abstract existence of 2600 cycles per second.

The same frequency may appear in music, speech, machinery, laboratory instruments, telecommunications testing, and natural sound. Its numerical value is not inherently criminal.

It becomes a lockpick when four conditions are combined:

the signal matches a machine-recognized command

the signal is introduced into a vulnerable system

the user lacks authorization

the user intends to manipulate the system or obtain an unlawful benefit

This combination shows why the phrase “illegal number” must always be interpreted relationally. The number identifies a frequency. The frequency produces a tone. The tone triggers a network response. The response becomes unlawful only in the context of unauthorized use.

The legal significance can be summarized as:

2600 Hz alone → neutral physical phenomenon

2600 Hz in a laboratory test → authorized technical signal

2600 Hz sent into a vulnerable trunk with permission → security research

2600 Hz sent to evade billing → fraudulent control input

The visible number remains unchanged. Authorization changes everything.

This framework closely resembles the treatment of passwords and cryptographic keys. Possessing one's own password is ordinary. Using another person's password without permission may be criminal. Publishing a key in a technical history differs from using it to defeat an access-control system.

The 2600-Hz case therefore supplies an early model for understanding operational strings. Their status depends not only on what they are, but on where, why, and by whom they are used.

5.6 The Birth of Machine-Interpreted Illegality

Telephone phreaking introduced a decisive change in the relationship between information and law. Earlier restricted numbers were interpreted mainly by human communities.

Pythagoreans interpreted irrationality as a challenge to philosophical order.

Hebrew readers interpreted certain letter combinations as sacred.

Roman authorities interpreted an imperial death date as political danger.

Florentine merchants interpreted zero and positional digits as potential instruments of fraud.

The telephone network did not interpret 2600 Hz symbolically. It responded mechanically.

The signal had no need to persuade a human operator. It caused equipment to change state because the system had been engineered to recognize that frequency.

This is the beginning of machine-interpreted illegality. The critical audience is no longer only a priest, ruler, merchant, judge, or censor. It is a technical system.

The machine does not ask whether the signal is sincere, ironic, artistic, historical, or politically justified. It reacts according to its design.

That property gives operational information unusual power. Human language is often ambiguous. Machine commands must generally satisfy specific structural conditions. Once those conditions are met, the system may act immediately.

The transition can be represented historically:

number as sacred symbol

number as political prediction

number as commercial record

number as machine command

This progression also changes the scale of possible consequences. A sacredly offensive notation affects a community of readers. A forged ledger entry may affect a financial transaction. A control signal can alter a nationwide network in real time.

Telephone phreaking anticipated several major principles of modern cybersecurity.

First, systems should separate control channels from user channels whenever possible.

Second, a valid command should not be trusted solely because it has the correct format.

Third, authorization must be authenticated independently of the command itself.

Fourth, information that appears harmless to humans may be operationally powerful to machines.

Telephone companies eventually moved toward out-of-band signaling systems in which control information traveled through channels inaccessible to ordinary callers. This architectural change reduced the effectiveness of classic blue boxes.

The solution was not to outlaw the number 2600. It was to redesign the system so that producing the tone no longer granted control.

This is a critical lesson for the entire history of illegal numbers. When a system treats a publicly reproducible string as sufficient authority, attempts to suppress the string will usually be less effective than changing the architecture.

A secret frequency can be discovered. A password can be leaked. A key can be copied. A jailbreak suffix can be redistributed. A restricted sequence can be re-encoded.

Security that depends entirely on preventing knowledge of a number is fragile.

The 2600-Hz era therefore forms the bridge between older symbolic restrictions and modern digital access control. It showed that a number could become dangerous not because people believed in its meaning, but because machines had been built to obey it.

6. Cryptography, Export Controls, and the Recognition of Source Code as Speech

Cryptography transformed the political meaning of executable information. A cryptographic algorithm could be written as mathematics, explained in prose, printed as source code, stored on a disk, compiled into software, or embedded in hardware. Each representation expressed essentially the same method, yet governments often treated them differently because some forms were easier for machines to execute.

During the late twentieth century, the United States regarded strong cryptography not merely as an academic subject or commercial technology but as a strategic capability with military and intelligence significance. Export controls restricted the international distribution of certain encryption systems, software, and technical information. These controls produced an unavoidable constitutional question: when source code explains an algorithm while also enabling a computer to perform it, is the code speech, machinery, or both?

The disputes involving Philip Karn and Daniel Bernstein exposed this problem with unusual clarity. Karn confronted a regulatory distinction between cryptographic code printed in a book and the same code stored electronically. Bernstein challenged licensing requirements that constrained publication of his encryption source code. Together, these controversies helped establish that source code has expressive qualities protected by the First Amendment, even though code can also operate machines.

This was a decisive development in the history of illegal numbers. Restrictions no longer concerned only secret credentials or unauthorized signals. They concerned publicly intelligible descriptions of methods. The legal system had to confront information that could teach, communicate, and execute at the same time.

6.1 Cryptography as a Military and National-Security Technology

Cryptography is the practice of transforming information so that only authorized parties can interpret or authenticate it. Its historical applications include military commands, diplomatic communications, intelligence operations, financial transactions, and protection of state secrets.

Because cryptography can conceal communications from governments as well as from criminals and foreign adversaries, states have often regarded it as a technology requiring control. Strong encryption can protect a nation's military communications, but it can also prevent that nation's intelligence agencies from reading the communications of others.

This dual role made cryptography strategically important. A country that possessed stronger encryption than its rivals could protect its plans while exploiting weaknesses in opposing systems. Cryptographic knowledge therefore belonged historically to the same world as signals intelligence, codebreaking, electronic warfare, and military communications.

During the Cold War, advanced cryptographic technology was frequently regulated through export-control systems associated with defense articles and munitions. The underlying assumption was that transferring strong encryption abroad could strengthen foreign military, intelligence, or terrorist capabilities.

This regulatory approach was easier to understand when cryptography was embodied in specialized machines. A cipher device could be manufactured, shipped, inspected, licensed, confiscated, or destroyed. The object had a physical location and an identifiable owner.

Software destabilized this model. An encryption program could be copied almost perfectly at negligible cost. It could be transmitted electronically, printed in a journal, dictated over a telephone, reconstructed from an academic paper, or independently implemented from a mathematical description.

The strategic capability no longer resided entirely in a machine. It resided in reproducible information.

This created a conflict between two institutional traditions. National-security regulation treated cryptography as controlled capability. Academic culture treated algorithms and source code as knowledge to be examined, criticized, improved, and published.

The growth of personal computers and electronic networks made the conflict unavoidable. Encryption was no longer limited to governments and large corporations. Ordinary citizens could use it to protect email, commercial transactions, medical records, political organizing, and personal files.

Cryptography was becoming infrastructure for civil society.

6.2 Algorithms, Implementations, and Machine Readability

A cryptographic algorithm is an abstract procedure. It can be expressed mathematically, described in prose, represented through diagrams, or written as executable instructions.

An implementation translates the abstract method into a form that a particular computer can perform. Source code is usually written so that human programmers can read and modify it. A compiler or interpreter then converts or processes that code into operations carried out by a machine.

These layers can be summarized as:

mathematical idea → algorithmic description → source code → executable behavior

The boundaries are not absolute. A sufficiently detailed mathematical paper may allow a competent programmer to create an implementation. Source code may be read as an explanation of the algorithm. Executable machine code can be analyzed to recover aspects of its design.

Export regulators nevertheless attempted to distinguish forms according to how immediately they could be used. Printed explanations were often treated more permissively than machine-

readable files. A book might require a reader to type, debug, and compile the code. A digital file could be copied and executed directly.

This distinction rested on a difference in friction rather than information. The intellectual content might be identical, but one representation imposed additional labor before it became operational.

The regulatory assumption was therefore:

greater machine readability → greater practical capability

That assumption had some practical basis. A disk containing functioning code was easier to use than hundreds of printed pages. Yet it also created conceptual instability. If two representations conveyed the same instructions, why should constitutional protection depend on the storage medium?

The problem became sharper as technology improved. Optical character recognition allowed printed text to be converted into digital files. Scanners reproduced books rapidly. Online communication transmitted images and documents across borders almost instantly.

Machine readability was becoming a temporary property rather than a stable category.

6.3 The Karn Controversy

Philip Karn, an engineer and computer-networking specialist, brought this contradiction into public view during the 1990s. He sought permission to export a book containing cryptographic source code and also to export an electronic version of code appearing in the book.

The printed book was treated as exportable published material. The electronic source code, despite conveying substantially the same information, was treated as regulated cryptographic software.

The result appeared paradoxical. Foreign readers could legally obtain the book, manually enter the code into a computer, and create the same software. What they could not receive as freely was a digital copy that spared them the labor of transcription.

The Karn controversy revealed that the government was not simply controlling knowledge of cryptography. It was controlling the convenience and immediacy with which the knowledge could be converted into functioning software.

This was an important distinction, but it was also difficult to defend as a durable legal boundary. A restriction based on typing effort becomes weaker as transcription becomes automated. Once a printed page can be scanned and converted accurately into source code, the difference between book and disk approaches zero.

The case also highlighted the geographical assumptions of export law. Traditional export controls were designed for physical goods crossing national borders. A crate of equipment could be inspected at a port. Digital information could cross borders through email, bulletin-board systems, file-transfer networks, or later the public internet.

Publication itself could become export.

The government's position therefore raised a broad question: could a person be required to obtain permission before communicating technical information to the world merely because foreign readers might receive it?

The issue extended beyond cryptography. It concerned the future of scientific communication in a networked society.

6.4 Bernstein and the First Amendment

Daniel Bernstein, then a graduate student, developed an encryption system and sought to publish the algorithm, explanatory materials, and source code. Federal export regulations required prior governmental authorization for some forms of cryptographic publication.

Bernstein challenged the restrictions on constitutional grounds. He argued that source code expressed scientific ideas and that requiring a license before publication amounted to an impermissible prior restraint on speech.

The litigation forced courts to examine source code not merely as a functional tool but as a language used by human beings. Programmers read source code to understand procedures, identify errors, compare methods, teach students, and communicate innovations. The code expresses relationships among variables, operations, data structures, and logical choices.

A program may therefore be studied even when it is never executed.

The recognition of source code as speech did not imply that every use of code was immune from regulation. Speech protections are not absolute, and conduct performed through language can still produce legal consequences. Fraud, threats, conspiracy, unauthorized access, and malicious computer activity do not become protected merely because they involve words or code.

The important principle was narrower and more foundational: source code does not lose its expressive character merely because a computer can act upon it.

This recognition prevented the government from treating all code as if it were indistinguishable from physical machinery. It also established that restrictions on publication required serious constitutional scrutiny.

The Bernstein litigation became a landmark in the development of digital free-speech doctrine. It established a framework for later disputes involving circumvention software, vulnerability research, digital keys, and instructions capable of producing regulated effects.

6.5 Source Code as Speech and Machine

Source code occupies an unusual position because it addresses two audiences simultaneously.

The first audience is human. Programmers can read code, discuss it, criticize it, and learn from it. Variable names, comments, structure, and organization convey meaning. Two implementations of the same algorithm may reflect different assumptions, priorities, and intellectual choices.

The second audience is computational. When processed by appropriate software and hardware, the code contributes to machine behavior.

This dual character can be expressed as:

source code → explanation for humans

source code → instruction for machines

Ordinary prose may also produce action, but usually through human interpretation. A written command influences a person who decides whether and how to obey. Source code can cause a machine to perform a sequence automatically once the necessary conditions are met.

This directness makes code appear more dangerous than conventional writing. Yet the distinction is one of degree rather than an absolute divide. Architectural plans guide construction. Chemical formulas enable synthesis. Musical notation directs performance. Legal documents transfer property. Written language has always had operational effects.

Source code intensifies this quality because execution can be exact, rapid, and scalable. One copy of a program can be run millions of times without further explanation from its author.

The legal challenge is therefore not to decide whether code is speech or machinery. It is to recognize that it is both.

A legal system that sees only speech may underestimate operational harm. A system that sees only machinery may suppress scientific discussion and technical criticism.

The appropriate analysis must consider context, intent, authorization, proximity to harm, and the availability of legitimate uses.

6.6 The Collapse of the Print-Digital Boundary

The distinction between printed and machine-readable information was already unstable during the encryption controversies of the 1990s. It has become still less defensible in the modern technological environment.

A smartphone can photograph a page and extract its text. Optical character recognition can convert printed source code into editable files. Language models can reconstruct missing syntax, correct transcription errors, translate code between programming languages, and generate implementations from prose descriptions.

A printed algorithm is therefore potentially executable information in temporary disguise.

The process may now require only a few steps:

photograph → extract → correct → compile → execute

A technical article without source code may still provide enough detail for automated systems to generate an implementation. A mathematical proof can be translated into software. A diagram can be interpreted as a circuit. A spoken explanation can be transcribed and converted into commands.

This technological convergence undermines regulatory systems that assign radically different legal status to equivalent representations.

It also complicates the idea of suppressing dangerous code. Removing a downloadable file may have limited effect if the method is explained in widely available publications. Conversely, restricting every description detailed enough to permit reconstruction would impose severe burdens on science, education, and journalism.

The problem is not solved by declaring all information harmless. Certain instructions can facilitate serious wrongdoing. The difficulty lies in regulating harmful conduct without imposing a general prohibition on understanding.

Modern AI makes this boundary even more porous. A reader no longer needs to possess extensive programming expertise to translate an explanation into code. Automated systems can assist with implementation, debugging, optimization, and adaptation.

The difference between knowledge and capability is narrowing.

6.7 Functional Speech as a New Legal Category

The encryption disputes helped define a broader category that may be called functional speech: expression that communicates ideas while also enabling or causing technical operations.

Functional speech includes source code, cryptographic keys, exploit instructions, smart-contract programs, machine-control sequences, and certain biological design files. These forms differ in their degree of directness, but all challenge the assumption that speech and action can be cleanly separated.

A useful framework distinguishes several levels of functionality.

Descriptive information explains how a system operates.

Instructional information teaches a person how to perform an operation.

Executable information can be processed directly by a machine.

Capability-conferring information grants access or authority, as with a password or private key.

Autonomous information continues producing effects after deployment, as with some smart contracts or self-propagating software.

The more directly information produces a prohibited consequence, the stronger the argument for narrowly tailored regulation. Yet direct functionality alone cannot decide the issue. The same encryption code may protect a dissident, secure a hospital, safeguard a bank, or conceal a criminal conspiracy.

Functionality does not equal illegitimacy.

The central legal question becomes whether regulation targets harmful conduct or suppresses knowledge because of its possible uses.

This distinction requires attention to several factors:

the information’s legitimate applications

the speaker’s purpose

the intended audience

the level of operational detail

the immediacy of potential harm

the existence of authorization

the availability of less restrictive safeguards

The cryptography controversies demonstrated that governments could not resolve these questions simply by classifying digital code as a weapon. Code was part of scientific and political

discourse. Restricting it affected not only national security but also privacy, commerce, journalism, and academic freedom.

The eventual liberalization of many encryption controls reflected technological reality as much as constitutional principle. Strong cryptography was spreading internationally, commercial demand was growing, and foreign implementations were widely available. Suppressing domestic publication could disadvantage local researchers and companies without preventing global access.

This pattern will recur in the history of illegal numbers. Once information is widely reproducible, attempts at prohibition may impose substantial costs on compliant users while doing little to prevent determined actors from obtaining it.

The encryption debates therefore form the intellectual bridge between telephone phreaking and the later controversies over DeCSS, illegal prime numbers, and leaked media-protection keys.

Telephone phreaking involved signals designed to control a network.

Cryptographic export disputes involved descriptions and implementations of methods designed to control information.

The next stage would combine both problems. Source code and numerical keys would not merely protect communication. They would unlock copyrighted media, defeat consumer access controls, and spread globally through the internet.

At that point, the conflict between code, speech, property, and regulation would become a mass public controversy.

7. Copyright Circumvention and the Golden Age of the Illegal Number

The late 1990s and early 2000s produced the most literal and widely recognized examples of "illegal numbers." Digital entertainment companies increasingly relied on encryption, authentication codes, and software restrictions to control access to movies, music, games, and operating systems. At the same time, the internet made it possible to distribute technical information to millions of people almost instantly.

This collision created a new legal problem. A cryptographic key might consist of only a few dozen hexadecimal characters. A decryption program could be converted into a large integer. A product key could fit on a handwritten note. Yet these compact strings could defeat technological restrictions governing valuable commercial products.

The central disputes were not really about mathematics. They concerned whether a sequence should be treated as speech, software, stolen property, a circumvention device, a credential, or some combination of all five.

The period from DeCSS through the AACS controversy may be called the golden age of the illegal number because the paradox became visible to the general public. Courts, corporations, programmers, activists, and internet users confronted a question that had previously seemed absurd: could publishing a number amount to distributing an unlawful machine?

The answer depended on context. No integer was unlawful in every form and setting. Yet a number could become legally actionable when it encoded software, supplied an access key, or formed part of a deliberate effort to defeat a protected technological system.

7.1 The Digital Millennium Copyright Act

The Digital Millennium Copyright Act of 1998 altered the legal landscape surrounding digital access controls. Traditional copyright law focused primarily on unauthorized reproduction, distribution, public performance, and the creation of derivative works. The new anti-circumvention provisions addressed an earlier stage: defeating the technological measures controlling access to a copyrighted work.

This distinction was significant. A person might circumvent an access control without immediately copying or distributing the protected work. The law nevertheless treated certain acts of circumvention and certain forms of trafficking in circumvention technology as independently actionable.

The statute addressed technologies, products, services, devices, components, and parts designed or marketed for bypassing access controls. In a digital environment, however, a “device” did not need to be a physical object. It could be a computer program. A program could be expressed as source code. Source code could be represented as binary data. Binary data could be converted into an integer.

The conceptual chain was therefore:

circumvention device → software → bits → number

This chain made the idea of an illegal number more than a rhetorical joke. If the law restricted distribution of a program because of its function, then encoding the same program numerically did not necessarily remove that function.

The DMCA was intended to give copyright owners confidence that technological protection systems would receive legal support. Without such support, a single widely distributed circumvention tool could undermine an entire commercial access-control scheme.

Critics argued that the law could also suppress legitimate security research, interoperability, preservation, criticism, and fair use. The controversy therefore involved competing claims about property, innovation, expression, and technological control.

7.2 DeCSS and DVD Encryption

Commercial DVDs commonly used the Content Scramble System, or CSS, to encrypt audiovisual data. Licensed DVD players contained the information needed to decrypt and play protected discs.

DeCSS emerged as software capable of descrambling CSS-protected DVD content. Its origins were associated with efforts to play DVDs on systems that lacked licensed playback software, including early Linux environments. Once released publicly, however, the program could also facilitate copying and redistribution.

Motion-picture companies and DVD licensing organizations sought to restrict publication of DeCSS and links directing users to it. Their legal arguments relied heavily on the DMCA's anti-circumvention provisions.

The disputes exposed the dual nature of source code. DeCSS could be read by programmers as a description of how DVD encryption worked. It could also be compiled and used to decrypt discs.

Supporters of publication framed the program as speech, research, and a tool for interoperability. Copyright owners framed it as a functional mechanism for bypassing access controls.

The legal conflict did not require a court to decide that all discussion of CSS was forbidden. Technical descriptions, criticism, and general information remained distinguishable from distributing a working program in circumstances intended to facilitate circumvention.

Yet the boundary was difficult to maintain. A sufficiently detailed explanation could allow a programmer to reconstruct the tool. A printed copy of the source could be typed into a computer. A numerical encoding could reproduce it exactly.

The effort to suppress DeCSS therefore generated experiments designed to show that executable information could inhabit almost any representational form.

7.3 The 1,401-Digit Illegal Prime

In 2001, programmer Phil Carmody created one of the most elegant demonstrations in the history of digital law. He encoded DeCSS-related source code into a 1,401-digit prime number.

The procedure depended on the fact that every finite digital file can be interpreted as an integer. A sequence of bits is already a binary number. That number can be written in decimal notation, tested for mathematical properties, or transformed into other forms.

Carmody selected and constructed a representation that was not merely an arbitrary large integer but a prime. The result could therefore appear in a mathematical database or discussion as an object of number theory while also containing recoverable program information.

The term “illegal prime” was intentionally provocative. No statute listed the prime itself as forbidden, and no court specifically ruled that the decimal number could never be published. The point was conceptual rather than procedural.

If the encoded program was unlawful to distribute in one form, did the same restriction apply when the program was written as a prime number?

If yes, then a mathematically interesting integer could become regulated because of its hidden interpretation.

If no, then the restriction could be defeated through a reversible change of notation.

The prime exposed the conflict between informational identity and physical representation. The law often focuses on the function or substance of a regulated object. Mathematics allows that substance to be transformed into infinitely many visible forms.

The same code might appear as:

decimal digits

binary digits

hexadecimal characters

compressed data

an image

a sequence of musical notes

a list of coordinates

a prime number

The underlying information remains recoverable when the decoding procedure is known.

The illegal prime therefore became a philosophical challenge to information regulation. It suggested that once software was treated as prohibited functionality, mathematics itself could become one of its containers.

7.4 Representation Invariance

Representation invariance is the principle that the functional identity of information can survive a change in notation.

A text file can be converted into binary. Binary can be written in hexadecimal. Hexadecimal can be mapped to colors. Colors can be stored in an image. The image can be converted back through the reverse sequence.

The outer appearance changes, but the recoverable informational content remains.

This creates a serious enforcement problem. A rule directed only at a specific character sequence is easy to evade. A rule directed at every equivalent representation becomes extraordinarily broad.

The set of equivalent representations is effectively unlimited. A prohibited program could be divided into pieces, encrypted, compressed, hidden in an unrelated file, expressed as a mathematical constant, or reconstructed from instructions.

This can be written abstractly as:

information I → encoding $E_1(I)$

information I → encoding $E_2(I)$

information I → encoding $E_3(I)$

If each encoding is reversible, all three can reproduce the same informational object.

The legal system must therefore decide what level of equivalence matters. Is a detailed description equivalent to working code? Is an encrypted copy equivalent if the key is published elsewhere? Is a photograph of a key equivalent to the key itself? Is a song that encodes the digits equivalent to a software component?

Absolute answers are difficult because context changes functionality. A decimal integer may be practically useless without a decoder. Yet when the decoder is simple and widely known, the separation may be artificial.

Representation invariance explains why digital suppression often fails. Removing one copy does not remove the information. It encourages transformation.

7.5 The AACS HD DVD and Blu-ray Processing Key

The controversy surrounding the Advanced Access Content System in 2007 brought the illegal-number problem to a much larger audience. AACS was used to protect high-definition optical media, including HD DVD and Blu-ray content.

A processing key capable of participating in the decryption of protected media was published online. The key was written as a short hexadecimal sequence. Unlike the 1,401-digit prime, it could be memorized, copied by hand, placed in a headline, or printed on clothing.

Rights-management organizations issued takedown demands to websites and services that displayed or linked to the key. These efforts treated the sequence as a component of a circumvention mechanism rather than as a meaningless string of characters.

The key's compactness made the controversy especially powerful. The allegedly restricted object was not a complex program spread across many pages. It was a small sequence that anyone could reproduce exactly.

The practical issue was more complicated than the popular slogan suggested. Possession of the processing key did not automatically provide effortless access to every protected disc. Decryption required software, technical knowledge, and additional system information. Keys could also be revoked or replaced in later media.

Nevertheless, the sequence demonstrated that a small amount of information could weaken a large technological-control system.

A security architecture involving corporations, licenses, encryption standards, hardware manufacturers, and copyright law had become vulnerable to a number short enough to fit on a T-shirt.

7.6 The Internet Revolt and the Streisand Effect

Attempts to remove the AACS key produced the opposite result. Internet users copied it across blogs, forums, news sites, social-media platforms, and collaborative communities.

One of the most dramatic conflicts occurred on Digg, a popular news-aggregation platform. The site removed posts containing the key and reportedly took action against some user accounts. Users responded by flooding the service with additional posts containing the same sequence.

The volume of republication made suppression impractical. Digg eventually reversed its position and allowed the key to remain.

The event became a classic example of the Streisand effect: an effort to suppress information draws attention to it and causes it to spread more widely.

Before the takedown campaign, the key was of interest mainly to technical communities. After the campaign, it became a symbol of internet resistance. Many people who had no intention of decrypting a disc copied the number as an act of protest.

The meaning of the key changed through circulation.

Initially, it was a technical credential.

Then it became alleged circumvention information.

Finally, it became a political slogan.

This transformation complicated the legal analysis. A person publishing the sequence as part of a functioning decryption package was doing something different from a person displaying it in a news report, legal argument, historical archive, or protest image.

The visible string was identical. The communicative purpose was not.

7.7 The Free Speech Flag

The Free Speech Flag provided one of the clearest demonstrations of representational transformation. Bytes associated with the AACS key were mapped to color values and displayed as a pattern of colored stripes or blocks.

To a casual viewer, the result appeared to be a simple abstract flag. To someone who knew the encoding scheme, the colors could be converted back into numerical information.

The flag asked a difficult question: if the hexadecimal text was restricted, were the corresponding colors also restricted?

The same question could be extended indefinitely. Would a photograph of the flag be restricted? What about a verbal description of the colors? What about a painting created from that description? What about a number representing the digital image of the painting?

The Free Speech Flag did not prove that every encoded form must receive complete legal immunity. It demonstrated that information regulation cannot be confined easily to surface appearance.

The flag also turned a technical dispute into political art. Its function was not primarily to help users decrypt media. It was to criticize the idea that a sequence of information could be removed from public discourse through legal demands.

This illustrates the capacity of a number to acquire multiple layers of meaning:

technical key

legal controversy

political symbol

artistic expression

historical artifact

A court or regulator assessing publication would therefore need to consider more than the raw digits. Purpose, audience, context, and practical effect would matter.

7.8 The PlayStation 3 Root-Key Dispute

The PlayStation 3 controversy extended the illegal-number problem into video-game consoles. Sony had designed the system with cryptographic protections intended to control which software could run on the hardware.

Security researchers and hackers identified weaknesses in the signing and verification system. Technical information, tools, and key material were subsequently published.

Sony brought legal action against individuals associated with the disclosures, including George Hotz and members of the fail0verflow group. The allegations involved more than the publication of a single number. They concerned reverse engineering, circumvention tools, unauthorized software, access to protected systems, and distribution of methods enabling users to bypass console restrictions.

The phrase “Sony made a number illegal” simplified a much broader dispute. The litigation did not establish a universal prohibition against writing a particular integer in every conceivable context.

Nevertheless, key publication played a central symbolic role. A cryptographic system designed to distinguish authorized from unauthorized software had been reduced to information that could be copied freely once discovered.

The case illustrated a recurring weakness in systems that place long-term trust in embedded secrets. Hardware may be physically controlled, but the secret values governing it can escape through flaws, reverse engineering, or implementation mistakes.

Once a master key becomes public, legal enforcement may slow its use but cannot restore its secrecy.

The technical response must therefore involve redesign, updates, revocation, or replacement. Litigation cannot make a widely published key unknown again.

7.9 The Windows XP FCKGW Product Key

The Windows XP product key beginning with the characters FCKGW became one of the most famous software credentials of the early personal-computing era.

The key was associated with a volume-license edition of Windows XP and reportedly leaked before or near the system’s commercial release. Because volume-license installations operated

differently from ordinary retail activation, the key could be used widely with compatible installation media.

Microsoft attempted to limit unauthorized use through blacklisting, service-pack checks, licensing changes, and later activation systems.

This example differs from DeCSS and the AACS key. The FCKGW string was not primarily a decryption algorithm or anti-circumvention program. It was a credential accepted by Microsoft's licensing system.

Its power came from institutional recognition. The characters did not perform complex mathematics. They caused software to treat an installation as authorized.

The product key therefore resembled a ticket, password, or account credential. Its value was relational:

string + Microsoft licensing system = accepted authorization

Once Microsoft changed the system so that the key was no longer accepted, the string lost much of its practical power.

This reveals an important difference between mathematical and institutional capability. The truth of √2 cannot be revoked. A product key can be invalidated because its function depends on a system continuing to honor it.

The FCKGW key belongs in the history of illegal numbers because it shows how a short alphanumeric sequence can acquire commercial authority. It also demonstrates that the most effective response to a leaked credential is usually revocation, not permanent suppression of the string.

7.10 Why No Integer Is Illegal in Every Context

The controversies of this period support a general conclusion: no ordinary integer or finite string is illegal in every possible context merely because it once functioned as a key, program, or circumvention component.

The same sequence may appear lawfully in:

a judicial opinion

a historical study

a cybersecurity course

a mathematical database

a news report

a museum exhibit

a technical archive

a criticism of copyright law

Its legal significance changes when it is integrated into conduct intended to defeat access controls, obtain unauthorized service, distribute copyrighted works, or facilitate infringement.

A useful distinction can be drawn among three activities:

describing a restricted technology

publishing information capable of implementing it

using or distributing it for unauthorized circumvention

The boundaries among these activities are not always clear. A description may be sufficiently detailed to become an implementation. Publication may be political commentary and practical assistance at the same time. A tool may have both lawful and unlawful uses.

The golden age of the illegal number revealed that digital law could not rely on simple categories inherited from physical property. A key is not exactly a crowbar. A program is not exactly a machine. A prime number is not exactly a DVD descrambler. Yet each can perform or encode functions traditionally associated with physical tools.

The central legal question is therefore not whether a number is abstract. All numbers are abstract. The question is whether a particular representation is being used as part of regulated conduct.

The DeCSS prime, AACS key, PlayStation 3 disclosures, and Windows XP product key each occupied a different position on this spectrum. One encoded software. One unlocked encrypted media. One participated in console-security circumvention. One authenticated software licensing.

Together, they established the modern image of the illegal number: a compact string situated at the intersection of mathematics, speech, property, and machine control.

They also revealed the practical limits of suppression. Once information has escaped into a global network, legal demands may remove individual copies but rarely eliminate the underlying knowledge. Takedowns can even increase public interest and generate new encodings.

The lasting lesson of this era is architectural. A security system that fails when one number becomes public is a fragile system. The long-term solution is not to make the number

unspeakable. It is to build systems that can survive disclosure, revoke compromised credentials, separate authority from mere knowledge, and minimize the power entrusted to any single secret.

The next stage in the history moves from machine control back into human politics. Numerical strings would become tools of censorship, protest, ideological signaling, and alleged threat. Unlike cryptographic keys, their force would depend not on exact machine execution but on contested social meaning.

8. Political Numbers, Historical Memory, Extremist Codes, and Alleged Threats

The modern political code returns the history of restricted numbers from machine interpretation to human interpretation. Unlike a telephone tone or cryptographic key, a political number normally does not cause machinery to perform a predetermined operation. Its power lies in what an audience understands it to mean.

A few digits can compress a date, slogan, legal provision, ideological doctrine, or implied command. Numerical compression offers several advantages. Numbers are short, visually memorable, easy to reproduce, and capable of passing through systems that search primarily for prohibited words. They may also provide plausible deniability. A speaker can claim that the digits were coincidental, humorous, ambiguous, or misunderstood, while an informed audience perceives a specific message.

This ambiguity makes political numbers difficult to regulate fairly. A code may serve as historical remembrance, protest, group identification, intimidation, extremist signaling, satire, or a genuine threat. The visible sequence alone rarely establishes which function is intended.

The examples considered in this section occupy different positions on that spectrum. The number 8964 preserves the memory of the Tiananmen Square crackdown under conditions of censorship. The number 1312 converts an anti-police slogan into alphabetic notation. The number 1488 signals white-supremacist and neo-Nazi ideology in many contexts. The number 187 invokes the California murder statute and can acquire threatening meaning when directed at a person. The expression 8647 combines political slang with the disputed interpretation of a message concerning the forty-seventh president of the United States.

These examples demonstrate that symbolic numbers are not ordinarily illegal in themselves. Their legal significance depends on context, audience, intention, institutional rules, and the likelihood that the communication will be interpreted as a serious call for action.

8.1 Numbers as Compressed Political Language

Political communities have always developed abbreviated forms of communication. Initials, colors, gestures, dates, songs, flags, and clothing can identify causes without stating them explicitly. Numerical codes belong to the same symbolic family.

Some political numbers are created through alphabet-position substitution. If A is one, B is two, and so forth, a slogan can be transformed into digits. Other codes refer to dates, penal statutes, historical events, leaders, ideological texts, or phrases counted by their number of words.

This compression can serve innocent and valuable purposes. A historical date can become a portable memorial. A protest slogan can fit on a wall, patch, or sign. A persecuted group can recognize its members without using terminology that would immediately attract censorship.

Compression can also serve harmful purposes. Extremist organizations can signal allegiance while avoiding explicit symbols. Gangs can threaten rivals through numbers understood within a local culture. Individuals can imply violence while preserving enough ambiguity to deny that any threat was intended.

A political code can therefore perform several functions at once:

It can identify the speaker.

It can identify the intended audience.

It can exclude outsiders who do not understand it.

It can condense a larger ideology.

It can evade automated moderation.

It can preserve deniability.

The meaning of a code is not contained entirely in the digits. It is distributed among the history of the symbol, the speaker's circumstances, the expectations of the audience, and the setting in which it appears.

A number painted on a history classroom wall may have a different meaning from the same number carved into a synagogue, posted beneath a politician's photograph, or included in a threatening message. Interpretation requires more than consulting a list of prohibited symbols.

This creates a general rule for political numerology:

sequence + shared cultural knowledge = symbolic message

The stronger the shared knowledge, the less explanation is required. Four digits can communicate an entire historical narrative to one audience while appearing meaningless to another.

8.2 The Meaning of 8964

The number 8964 is a compressed reference to 1989, the sixth month, and the fourth day. It points to June 4, 1989, the date most closely associated with the Chinese government's military suppression of protests centered on Tiananmen Square in Beijing.

The sequence is powerful because it preserves historical memory in an extremely compact form. It can be written as 8964, 89-6-4, 6/4/89, or through related combinations. Each version directs informed readers toward the same event.

The number does not contain a description of the demonstrations, the military response, the casualties, or the subsequent censorship. Its meaning depends upon the historical knowledge supplied by the audience.

This makes 8964 a numerical memorial. It allows people to remember and refer to an event even when direct descriptions are restricted.

The sequence also demonstrates the limitations of censorship. A government can prohibit phrases, remove photographs, block searches, control textbooks, and punish public commemorations. Yet each prohibition encourages the creation of substitute expressions.

If "Tiananmen Square" is filtered, users can refer to June Fourth.

If "June Fourth" is filtered, they can use 8964.

If 8964 is filtered, they can use related dates, images, mathematical expressions, inverted characters, candle symbols, or indirect historical references.

The censorship target therefore expands from a phrase to an equivalence class of representations. The more substitutes users invent, the more complicated enforcement becomes.

8.3 Censorship, Search Filtering, and Platform Enforcement

It is imprecise to describe 8964 simply as an illegal integer. The sequence has not been removed from arithmetic, and it may appear innocently as a password fragment, inventory number, price, year designation, or part of a larger calculation.

The restriction concerns political interpretation. When the number appears near the June Fourth anniversary, in a discussion of political history, or in a commemorative setting, authorities and platforms may recognize its intended meaning.

Enforcement can take several forms:

search results may be blocked;

social-media posts may be removed;

accounts may be suspended;

product-personalization systems may reject the sequence;

public displays may attract police attention;

commemorative activity may be restricted.

These measures differ from a single criminal statute declaring that no person may write 8964. They form a distributed system of information control involving government agencies, private platforms, automated filters, human moderators, commercial firms, and public-security practices.

This distributed structure makes censorship difficult to observe. A user may not know whether a search produces no result because no content exists, because the platform filtered it, or because the original publisher deleted it. A rejected engraving request may appear to be a technical error rather than political censorship.

Numerical censorship can therefore operate silently. A prohibited phrase attracts attention because its removal is visible. A blocked search term may simply produce an empty screen.

The treatment of 8964 reveals that political restriction is increasingly implemented through infrastructure. The state does not need to prosecute every person who writes a code. It can pressure the systems through which the code would circulate.

8.4 The Meaning of 1312

The number 1312 is commonly derived by substituting the alphabetical positions of the letters in ACAB:

A = 1

C = 3

A = 1

B = 2

The resulting sequence represents an anti-police slogan. It appears in graffiti, tattoos, clothing, music, demonstrations, stadiums, online posts, and political subcultures.

Its meaning is generally oppositional rather than secret. Many people who use 1312 do so precisely because they expect others to recognize the underlying phrase. Numerical substitution makes the slogan compact and may help it evade rules that prohibit the explicit wording.

The code can express several related attitudes. It may condemn police institutions, protest misconduct, reject policing as a social system, express hostility toward particular officers, or signal identification with an anti-authoritarian subculture.

These meanings should not be treated as identical. A generalized criticism of an institution differs legally and morally from a personal threat against an identifiable officer.

The same sequence may also appear without political meaning. It can represent a year, address, room number, model designation, or random part of a larger number. Context remains essential.

8.5 Contextual Enforcement in Europe

It is inaccurate to say that 1312 is uniformly illegal throughout Europe. European countries have different laws governing insult, public order, incitement, extremist propaganda, stadium conduct, vandalism, and speech directed at public officials.

Even within one jurisdiction, courts may distinguish between criticism of the police as a broad institution and an insult directed at a small, identifiable group of officers.

A banner displayed to thousands of spectators may express a generalized political position. The same words shouted at a particular officer during a confrontation may be treated as a personal insult. Graffiti on private property may be prosecuted as vandalism regardless of whether the slogan itself is protected.

The legal analysis therefore involves at least two separate questions:

Is the message itself punishable?

Was the method of displaying it independently unlawful?

A person may have the right to express a political opinion but not the right to damage a building while doing so. Conversely, the absence of property damage does not automatically protect speech that constitutes a true threat or targeted harassment.

The 1312 example demonstrates why reports that a number “carries a fine” must be interpreted cautiously. A fine may arise from local public-order rules, an insult statute, stadium regulations, vandalism, or another offense rather than from the abstract sequence itself.

8.6 The Meaning of 1488

The number 1488 combines two widely recognized white-supremacist codes.

The first component, 14, refers to a fourteen-word white-supremacist slogan. The second component, 88, commonly represents HH because H is the eighth letter of the alphabet. In neo-Nazi usage, HH abbreviates a salute to Adolf Hitler.

When used together, the components can signal adherence to white-supremacist or neo-Nazi ideology.

The number appears in tattoos, usernames, graffiti, merchandise, organizational names, propaganda, prison communications, and personalized identifiers. It is useful to extremists because it communicates allegiance without requiring an explicit swastika or written declaration.

Yet 1488 cannot be interpreted mechanically in every setting. It may appear as a year, price, address, historical statistic, document number, or randomly assigned code. A person born on January 4, 1988, for example, might produce a similar sequence without ideological intent.

Even 88 alone is highly ambiguous. It may be a jersey number, year, radio expression, address, or ordinary quantity. The extremist meaning becomes stronger when it appears beside other symbols, slogans, names, or behavior associated with white supremacy.

Recognition must therefore be probabilistic rather than absolute. The code is evidence of ideology in some contexts, not proof in every context.

8.7 License Plates and Administrative Symbol Control

Personalized license plates create a special problem because they combine private expression with government-issued identification.

A driver chooses the requested sequence, but the state manufactures and displays the plate under an official registration system. Authorities therefore review proposed combinations and may reject those associated with obscenity, threats, criminal organizations, or extremist ideology.

Some European jurisdictions restrict license-plate combinations associated with Nazism. These policies do not prohibit historians from printing the numbers, mathematicians from calculating with them, or journalists from discussing their meaning. The restriction applies to an official public identifier.

The administrative rationale is that the state should not appear to approve, reproduce, or distribute extremist symbolism through its own registration system.

The policy also creates difficult classification problems. Automated lists can reject innocent combinations. Human reviewers may fail to recognize new codes or may interpret ambiguous numbers inconsistently.

License-plate regulation therefore illustrates a narrower form of numerical control. The number is not banned from society. It is excluded from a particular state-managed channel.

This distinction between general prohibition and channel-specific restriction will recur in digital platforms. A company may refuse a username or engraving without claiming that the sequence is unlawful to write elsewhere.

8.8 The Meaning of 187

Section 187 of the California Penal Code defines murder. Through music, gang culture, films, police language, and popular media, the number 187 became a shorthand reference to killing.

Its meaning differs from the ideological structure of 1488 or the memorial function of 8964. It originates in a legal citation and then migrates into slang.

This migration demonstrates how law can generate symbols beyond the courtroom. A penal-code number intended for lawyers and judges becomes a cultural expression understood by audiences who may never have read the statute.

The sequence can appear descriptively. A journalist may discuss Section 187. A musician may employ it as part of fictional storytelling. A law student may write it in class. In such contexts, the number is not a threat.

Its significance changes when directed at an identifiable person. If 187 is written beside a person's name, photograph, address, vehicle, or workplace, it may communicate an intention or wish that the person be killed.

The legal inquiry then extends beyond the number:

Who produced the message?

Who was targeted?

What relationship existed between the parties?

Were weapons, surveillance, stalking, or prior threats involved?

Would the intended recipient understand the code?

Did the speaker intend the message to be received as a serious threat?

The number supplies vocabulary, but context supplies force.

8.9 The 8647 Controversy

The expression 8647 combines two disputed components. The number 47 can refer to Donald Trump as the forty-seventh president of the United States. The meaning of 86 is less stable.

In restaurant and bar usage, to "86" an item commonly means that it is unavailable, should be removed from the menu, or should no longer be served. Applied to a person, the term may mean to eject, remove, exclude, or get rid of that person.

Some speakers also use or interpret 86 in a violent sense. The degree to which that meaning is conventional, widespread, or obvious is disputed.

This ambiguity lies at the center of the 8647 controversy. One interpretation is political: remove the forty-seventh president from office or political life. Another interpretation is violent: kill or physically harm him.

A criminal prosecution cannot resolve that dispute merely by demonstrating that the violent interpretation exists. It must address whether the communication, viewed in context, constituted a serious expression of violent intent and whether the defendant possessed the mental state required by law.

The example is therefore more complex than a numerical dictionary entry. Slang has no central authority. Meanings vary by region, generation, occupation, and community. A term may begin with one meaning, acquire additional meanings, and retain ambiguity among them.

Political polarization further affects interpretation. Supporters of a public figure may perceive a hostile code as a threat. Opponents may perceive it as ordinary political rhetoric. Neither reaction alone determines the legal result.

8.10 The James Comey Prosecution

In May 2025, former Federal Bureau of Investigation Director James Comey posted a photograph showing seashells arranged as "86 47." The post generated immediate controversy.

Officials and supporters of President Donald Trump interpreted the arrangement as a threat against the president. Comey removed the image and denied intending violence, explaining that he had understood the arrangement as a political message rather than a call for physical harm.

After a federal investigation, a grand jury in April 2026 indicted Comey on charges alleging a threat against the president and transmission of a threatening communication in interstate commerce.

An indictment is not a conviction. It establishes that a grand jury found probable cause to permit the prosecution to proceed. The government still bears the burden of proving every element of the charged offenses beyond a reasonable doubt.

Comey remains entitled to the presumption of innocence. His denial of violent intent, the ambiguity of the term 86, the political context, the form of the image, and the reactions of viewers are among the issues likely to matter in evaluating the communication.

The prosecution is historically important because it places the concept of a dangerous numerical code directly before a federal criminal court. Unlike 2600 Hz, the sequence did not command machinery. Unlike a private encryption key, it did not unlock a protected system. Its alleged danger depended on human interpretation.

The case therefore represents a return to the ancient political problem discussed in the Roman section. A compact numerical expression becomes legally significant because authorities believe it concerns the removal or death of a ruler.

The constitutional environment is radically different, however. An American prosecution must operate under the First Amendment, the statutory elements of the charged offenses, the burden of proof, judicial review, and the requirement of a constitutionally sufficient mental state.

8.11 True Threats and Mental State

The First Amendment protects harsh criticism, political opposition, offensive language, satire, hyperbole, and many forms of emotionally charged rhetoric. It does not protect every communication associated with violence.

A true threat is a serious expression conveying that violence will be committed against a person or group. The doctrine protects individuals from fear and disruption while attempting to avoid criminalizing political exaggeration, jokes, artistic expression, or misunderstood language.

Modern constitutional doctrine requires more than asking whether an objectively reasonable listener could interpret the words as threatening. The speaker's mental state also matters.

At minimum, the government must establish a constitutionally sufficient degree of awareness that the communication would be understood as threatening violence. Recklessness, in this context, involves conscious disregard of a substantial risk concerning the threatening nature of the communication.

This requirement is essential in coded-speech cases. Ambiguous numbers may produce widely varying interpretations. Without a mental-state requirement, a speaker could face criminal punishment whenever an audience supplied the most threatening available meaning.

Several forms of evidence may bear on mental state:

the speaker's prior statements;

knowledge of the code's possible meanings;

the identity and vulnerability of the target;

the timing of the communication;

private messages or planning;

repetition after warnings;

efforts to conceal or explain the message;

the existence of weapons or preparations;

the reaction the speaker appeared to seek.

No single factor necessarily controls. Deleting a message, for example, might indicate consciousness of guilt, recognition of misunderstanding, fear of controversy, or a desire to reduce danger. Interpretation requires the totality of circumstances.

8.12 The Problem of Ambiguous Numerical Speech

Ambiguous numerical speech creates a conflict between two serious concerns.

The first is public safety. Coded language can allow genuine threats, extremist recruitment, criminal coordination, and intimidation to occur behind apparently innocent symbols. Authorities cannot ignore a message merely because it uses numbers rather than explicit words.

The second is freedom of expression. Political rhetoric often employs metaphor, humor, slang, exaggeration, and symbolic shorthand. Treating every hostile or ambiguous code as a criminal threat would chill lawful opposition and give governments excessive power to punish critics.

The solution cannot be a universal list of illegal numbers. Symbolic meaning changes too rapidly and depends too heavily on context.

A more defensible analysis asks:

Is the code's meaning established within the relevant audience?

Was there an identifiable target?

Did the communication express or imply violence rather than political removal?

Was the message part of a pattern?

What did the speaker understand the code to mean?

Did the speaker consciously disregard the risk that it would be received as a serious threat?

Was there evidence of preparation, capability, or coordination?

Could the communication reasonably be understood as satire, reporting, historical discussion, or criticism?

These questions do not guarantee agreement. Symbolic cases are difficult precisely because the same expression can carry several meanings at once.

Political numbers occupy an unstable position between language and action. They can preserve forbidden history, challenge institutions, signal ideology, mark group membership, frighten targets, or coordinate violence. Their brevity magnifies their ambiguity.

This section also reveals a progression within political coding itself.

8964 points backward toward a suppressed event.

1312 expresses opposition to an institution.

1488 signals ideological affiliation.

187 imports the language of murder into interpersonal communication.

8647 becomes the subject of a prosecution alleging a threat against a sitting president.

The sequence moves from memory to protest, from identity to intimidation, and from disputed symbolism to criminal accusation.

Yet the governing principle remains consistent. The digits alone are insufficient. A number becomes politically actionable only through an interpretive community and a surrounding course of conduct.

This principle distinguishes symbolic codes from the operational credentials discussed in the following sections. A password or private key does not require broad cultural agreement. The relevant system has been engineered to recognize it. Political numbers, by contrast, exist within contested human language.

Their power depends on who sees them, what history they recall, what action they imply, and whether the speaker intends the audience to understand them as memory, resistance, allegiance, or menace.

9. Credentials, Account Numbers, Passwords, and Numbers That Confer Authority

The modern information economy depends on strings that do more than describe people, accounts, or machines. They confer authority. A password can open an email account. A personal identification number can authorize a banking transaction. A credit-card number can initiate payment. A cryptographic private key can transfer digital assets. A reset token can replace an existing credential and assume control of an identity.

These strings are not powerful because of their mathematical properties alone. Their authority arises from institutions and technical systems that agree to recognize them. A bank, computer server, licensing system, or blockchain treats possession of the correct sequence as evidence that the presenter is entitled to act.

This creates one of the clearest modern examples of a number becoming legally restricted. The abstract digits remain lawful. What matters is the relationship among the sequence, the protected account, the person possessing it, and the intended use.

The same credit-card number may be lawful in the hands of the cardholder, necessary in the records of a payment processor, sensitive in a merchant database, stolen in a criminal marketplace, and evidentiary in a prosecution. Nothing about the digits themselves changes. Their legal status changes because authorization changes.

Credentials therefore show that illegal numbers are relational objects. They cannot be understood without asking who issued them, what system recognizes them, who possesses them, how they were obtained, and what actions they enable.

9.1 Access Devices in Modern Law

Modern law often avoids limiting fraud statutes to physical cards or mechanical keys. It uses broader concepts such as access devices, authentication information, account credentials, passwords, codes, and identifying numbers.

An access device may include a credit card, debit card, account number, personal identification number, electronic serial number, mobile identification number, password, token, or another means of obtaining money, goods, services, or access to an account.

This broad treatment reflects technological change. A criminal no longer needs to steal a physical wallet to misuse a financial account. The essential information may be copied from a database, intercepted during transmission, photographed, purchased from a criminal marketplace, or obtained through deception.

The law therefore focuses increasingly on capability rather than material form.

A plastic card is only one container for account information. The same information may exist as:

digits embossed on a card;

data encoded in a magnetic stripe;

information stored in a chip;

a token stored in a mobile wallet;

a record in a merchant database;

a photograph;

a handwritten note;

a message transmitted between criminals.

The physical card can remain in the owner's possession while the functional credential has been stolen.

This is a major departure from ordinary property crime. A stolen automobile is removed from its owner. A stolen number can be duplicated while the original remains visible and apparently unchanged. Both the authorized user and the thief may possess the same sequence at the same time.

Digital theft is therefore often non-rivalrous. Copying does not deprive the original possessor of the information, but it deprives that person of exclusive control and creates the risk of unauthorized action.

Possession alone may not always establish criminal liability. Context, quantity, method of acquisition, knowledge, and intent matter. A merchant may lawfully process thousands of account numbers. A security researcher may encounter leaked credentials while investigating a breach. A journalist may receive documents containing sensitive data. A criminal may possess the same information for fraudulent use.

The law must distinguish among these situations without pretending that the number itself determines guilt.

9.2 Descriptive Strings and Capability Strings

A useful distinction separates descriptive strings from capability strings.

A descriptive string tells a person or system what something is or where it can be found. An account identifier may identify a customer. A public blockchain address identifies a destination. A vehicle identification number identifies a specific automobile. A file hash identifies digital content.

A capability string enables an action. A password permits access. A private key authorizes a digital signature. A session token allows continued use of an authenticated service. A reset code permits replacement of a password. A one-time code confirms a transaction.

The distinction is not always absolute. An account number may be primarily descriptive in one system and sufficient to initiate transactions in another. A public address may be harmless by itself but become sensitive when combined with additional personal information.

The difference can be summarized as:

descriptive string → identifies an object, person, account, or location

capability string → authorizes or enables an operation

Capability strings are particularly sensitive because knowledge can substitute for identity. The system may not know the human being presenting the credential. It knows only that the correct sequence has been supplied.

This is the operational logic behind many authentication systems:

correct credential → presumed authority

The presumption is convenient but dangerous. If the credential is copied, guessed, intercepted, or coerced, the system may grant authority to the wrong person.

The legal consequences therefore arise from the gap between apparent authority and actual authorization. A computer may accept a password while the law regards its use as unauthorized.

Machines validate syntax and cryptographic correctness. Courts must determine entitlement.

9.3 Bearer Capabilities

Some credentials operate like bearer instruments. Whoever possesses them can exercise the associated capability, regardless of how possession was obtained.

Cash is a traditional bearer asset. A banknote generally does not identify its lawful owner. Physical possession allows the holder to spend it.

Certain digital credentials operate similarly. A private cryptographic key may allow its possessor to transfer cryptocurrency, sign software, decrypt messages, or authenticate as the controller of an account. The relevant system may have no independent method for distinguishing the rightful owner from a thief.

This produces the practical rule:

control of key = control of capability

The rule is not always legally true, but it may be technically true. A thief who steals a private key does not become the lawful owner of the assets, yet the blockchain may still execute the thief's transaction.

Bearer capabilities are powerful because they compress authority into information. A long legal relationship involving ownership, contract, identity, and permission may be represented operationally by a short string.

This compression creates extraordinary efficiency. Transactions can occur automatically across global networks without requiring human approval at every stage.

It also creates concentrated risk. A single copied secret may allow irreversible action.

Examples include:

a cryptocurrency private key;

an administrator password;

a cloud-service access token;

a software-signing key;

a password-reset link;

a recovery seed phrase;

an authentication cookie.

A bearer capability may be more dangerous than a descriptive database record because it does not merely reveal information. It allows the possessor to act.

The law struggles with this distinction when determining damages. Was the credential merely exposed, or was it used? Was there unauthorized access, attempted access, transfer of assets, or continuing risk? Did the system revoke the credential before harm occurred?

The answers affect whether a leaked string is treated as sensitive information, stolen property, an attempted instrument of fraud, or evidence of completed intrusion.

9.4 Authorized Possession and Criminal Possession

The same credential may be held simultaneously by several authorized parties. A bank retains account information. A customer possesses a card. A merchant temporarily processes payment details. A payment network routes the transaction. A fraud-monitoring service analyzes patterns associated with the account.

Possession is therefore not inherently suspicious.

Criminality generally arises from additional elements such as unauthorized acquisition, fraudulent intent, trafficking, use without permission, or knowing participation in a scheme.

The distinction may be expressed as:

credential + authorization → ordinary access

credential + mistake → possible negligence or accidental exposure

credential + unauthorized acquisition → compromise

credential + fraudulent intent → potential criminal instrument

credential + unauthorized use → completed access or fraud

This framework explains why context matters more than the visible number.

A person who finds a credit card and returns it has temporarily possessed an access device without becoming a fraudster. A system administrator may hold password hashes as part of legitimate employment. A forensic investigator may examine stolen credentials as evidence.

By contrast, possession of large collections of account numbers accompanied by expiration dates, security codes, victim information, and tools for testing transactions may strongly indicate fraudulent purpose.

Quantity can matter because it affects inference. One number may have been received accidentally. Thousands of organized credentials are more difficult to explain innocently. Yet quantity alone should not determine guilt, because legitimate organizations also process enormous datasets.

Authorization remains the central issue.

This creates a difference between technical access and legal access. A person may have the technical ability to enter a system because a password works. That does not establish legal permission. Conversely, a person may have legal permission but be unable to access the system because a credential has expired.

Capability and authority are related but distinct.

9.5 The Changing Status of a Credential

A credential has a life cycle. It may begin as valid and confidential, become exposed, be revoked, expire, and later survive only as historical or evidentiary information.

A simplified sequence is:

issued → activated → used → exposed → revoked → archived

At each stage, the same string has a different practical and legal status.

When issued, the credential may confer active authority.

When exposed, it becomes a security risk.

When revoked, it may lose operational power.

When archived, it may remain important for investigation or historical analysis.

This temporal character distinguishes institutional credentials from mathematical truths. The value of √2 does not expire. A product key, password, or authentication token can become useless when the issuing system stops accepting it.

Revocation is one of the most important security responses to disclosure. If a secret has become public, no legal order can make every observer forget it. The system must stop trusting it.

Effective systems therefore assume that credentials may eventually leak. They limit the scope, duration, and authority of each credential.

A strong design may include:

expiration dates;

one-time codes;

transaction limits;

device binding;

multifactor authentication;

rapid revocation;

separate permissions for separate tasks;

monitoring for unusual use.

These mechanisms reduce dependence on perpetual secrecy.

The changing status of a credential also affects publication. Printing an active password may facilitate immediate intrusion. Printing the same password years after revocation may serve historical scholarship without enabling access.

A legal analysis that ignores time may treat these situations as equivalent even though their practical consequences differ greatly.

9.6 Hash Values as Numerical Fingerprints

A cryptographic hash function converts input data into a fixed-length output. The output is usually displayed as a sequence of hexadecimal characters, although it is fundamentally a number represented in a convenient notation.

The process can be written conceptually as:

file → hash function → hash value

A small change to the input normally produces a very different output. This property makes hashes useful for checking integrity and identifying known files.

Suppose investigators calculate the hash of a seized file and compare it with a database of hashes associated with known malware or contraband material. A match may indicate that the file is identical to a previously classified object.

The hash acts as a fingerprint.

Hash matching offers practical advantages. Investigators can identify known content without manually opening every file. Companies can block repeated uploads of previously identified material. Software distributors can allow users to verify that a downloaded file has not been altered.

The hash itself is generally not the underlying file. It is a compact derivative identifier.

This distinction is crucial:

file contains content

hash identifies or verifies content

A conventional hash does not ordinarily provide enough information to reconstruct a large original file. Knowing the fingerprint of a book does not provide the text of the book.

For this reason, possession of a hash associated with prohibited content is not normally equivalent to possession of the content itself. Investigators, researchers, courts, and technology providers may lawfully maintain hash databases for detection and prevention.

The hash becomes legally significant as evidence. It may support the claim that a file found on one device is identical to a known file found elsewhere. It may help establish distribution, duplication, or prior classification.

Yet evidentiary use requires care. The reliability of the hash algorithm, the integrity of the database, chain of custody, classification procedures, and possibility of error all matter.

9.7 A Hash Is Not the Underlying File

The distinction between identifier and object prevents a major conceptual error in discussions of illegal numbers.

A number may point to illegal material without containing that material.

A library catalog number identifies a book but does not reproduce its pages. A vehicle identification number identifies an automobile but does not contain the automobile. A cryptographic hash identifies digital content but does not usually encode a reversible copy.

This differs from the DeCSS prime discussed earlier. The large prime was deliberately constructed so that the underlying program could be recovered through a decoding process. It functioned as a container.

A conventional hash functions primarily as a fingerprint.

The contrast can be stated as:

reversible encoding → information can be reconstructed

one-way hash → information can be compared but not ordinarily reconstructed

This difference matters for legal classification. A reversible encoding may amount to distribution of the underlying information. A non-reversible identifier generally does not.

The same principle applies to financial identifiers. Publishing a public blockchain address does not reveal the private key controlling it. Publishing the last four digits of a credit card does not normally allow use of the account. Publishing a username does not disclose the password.

Not all numbers associated with a capability possess that capability.

Careful analysis must therefore ask what can actually be done with the sequence. Does it authorize an operation? Does it reconstruct prohibited content? Does it merely identify something? Does it require additional secrets?

Without this functional inquiry, law and policy may exaggerate the danger of harmless identifiers while overlooking the power of seemingly innocuous tokens.

9.8 False Matches, Secret Lists, and Procedural Fairness

Automated numerical matching creates risks as well as benefits. Governments and companies increasingly rely on hashes, account identifiers, device numbers, risk scores, and credential databases to detect prohibited activity.

These systems can make decisions at enormous scale. A transaction may be blocked because an account matches a sanctions list. An upload may be removed because its hash matches a prohibited-content database. A login may be rejected because a device identifier appears suspicious.

The affected person may receive little explanation beyond a generic message.

This creates several procedural concerns.

First, databases can contain errors. An identifier may be entered incorrectly, associated with the wrong person, or remain listed after circumstances change.

Second, approximate matching can produce false positives. Exact cryptographic hashes are highly specific, but perceptual or similarity-based systems estimate resemblance rather than identity.

Third, lists may be secret. A user may be unable to inspect the evidence, understand the classification, or challenge the result.

Fourth, identifiers can be reassigned or shared. An internet address may later belong to a different user. A telephone number may be recycled. A device may be sold. An account may be compromised.

Fifth, an identifier does not prove who performed an action. A transaction associated with an account may have been initiated by an authorized user, employee, thief, malware program, or automated service.

These limitations show why numerical evidence must not be mistaken for complete proof.

A match is often the beginning of an investigation rather than its conclusion.

Procedural fairness requires mechanisms for review, correction, appeal, and contextual interpretation. Systems should distinguish high-confidence exact matches from uncertain similarities. They should preserve audit trails and avoid treating secret classifications as infallible.

The central danger is that numerical systems appear objective. A database match looks precise because it involves exact characters. Yet the meaning assigned to the match depends on human decisions about collection, labeling, thresholds, and consequence.

Numbers can conceal judgment rather than eliminate it.

The history of credentials and identifiers therefore reveals two opposing tendencies.

On one side, short strings make modern life possible. They allow rapid authentication, payment, communication, and access across global networks.

On the other side, they concentrate authority and enable automated exclusion. A copied credential can transfer control. A mistaken identifier can block an innocent user. A secret numerical list can impose punishment without transparent process.

The governing question is not whether the number is correct as a sequence. It is whether the institution's interpretation of that number is correct.

This section also clarifies the difference between several categories of information:

A password grants access.

An account number identifies or facilitates a transaction.

A private key authorizes cryptographic action.

A revoked credential preserves historical evidence but no longer grants authority.

A hash identifies content without ordinarily reproducing it.

A database match triggers suspicion but does not establish guilt by itself.

These distinctions will become especially important in the next section. Cryptocurrency systems combine public identifiers, bearer capabilities, financial assets, sanctions, and autonomous code. A blockchain address may identify a person, hold property, receive funds, or point to a program that executes without human intervention.

The evolution from passwords to smart contracts marks another increase in the causal power of numerical strings. A credential no longer merely opens a door. It may transfer wealth or activate a process that no one can later stop.

10. Cryptocurrency Addresses, Sanctions, and Autonomous Code

Cryptocurrency systems combine several categories of restricted numbers within a single technological environment. A public address can identify a destination for funds. A private key can authorize transfers. A transaction hash can identify an event. A smart-contract address can provide access to an executable process. Government sanctions lists can associate particular addresses with blocked persons, criminal organizations, state-sponsored hackers, or illicit financial activity.

These strings resemble ordinary account numbers, but their operation differs from conventional banking. A bank can freeze an account by changing an entry in its internal database. A

blockchain is maintained across a distributed network whose participants follow shared rules. No single institution necessarily possesses the power to erase an address, reverse a transaction, or switch off deployed code.

This creates an important distinction between regulating people and regulating protocols. Governments can prohibit individuals and businesses under their jurisdiction from engaging in specified transactions. They can sanction organizations, seize servers, prosecute developers, close websites, and require regulated exchanges to block assets. They may be unable, however, to eliminate an autonomous program already replicated across a public blockchain.

The Tornado Cash litigation brought this problem into sharp focus. Some of the addresses placed under sanctions did not simply identify conventional accounts controlled by an owner. They identified immutable smart contracts that continued operating without an administrator capable of changing or terminating them.

The dispute therefore asked whether a string could be more than an identifier. Could an address be the entrance to an autonomous machine? If so, could a legal system designed to block property controlled by persons also block software that no person owned or controlled?

10.1 Public Addresses and Private Keys

A cryptocurrency address is an alphanumeric identifier used within a blockchain system. It commonly serves as a destination to which digital assets can be transferred. Different blockchains use different address formats, lengths, encoding methods, and verification rules.

An address is often compared with a bank-account number, but the analogy is incomplete. A bank account exists inside the records of a financial institution. The bank determines balances, authorizes transfers, verifies identities, and can freeze or close the account.

A blockchain address exists within a distributed ledger system. The ledger records transactions associated with addresses according to the network's protocol. The assets are not physically contained inside the visible string. Rather, the blockchain records the state of asset ownership or control in relation to that address.

A public address may be shared freely. A person can publish it on a website, include it in an invoice, place it in a quick-response code, or send it to someone who wishes to make a payment.

The private key performs a different role. It is secret cryptographic information used to generate a valid digital signature. That signature demonstrates to the network that the transaction was authorized by whoever controls the relevant key.

The relationship can be simplified as:

public address → identifies a destination or ledger relationship

private key → authorizes movement or control

The public address is therefore generally descriptive. The private key is capability-conferring.

Publishing a public address does not normally allow someone to transfer assets away from it. Publishing the corresponding private key may allow anyone who obtains it to exercise the associated authority.

This is why saying that cryptocurrency is “stored in a wallet” can be misleading. A wallet commonly stores or manages keys. The blockchain maintains the transaction record. Losing the wallet application may not destroy access if the key or recovery phrase survives. Losing every copy of the private key may make the associated assets practically inaccessible even though the ledger continues to display them.

The distinction also clarifies the legal meaning of sanctioned addresses. Listing a public address does not necessarily place a secret key in government possession. It identifies a blockchain location or transactional relationship believed to be associated with a sanctioned actor.

The number has evidentiary and compliance value even though it does not itself confer control.

10.2 Cryptocurrency Addresses in Sanctions Records

Economic sanctions traditionally identify persons, companies, vessels, aircraft, bank accounts, and other property associated with designated actors. Cryptocurrency required sanctions authorities to add a new category of identifier: the digital-currency address.

A listed address can help exchanges, payment processors, financial institutions, investigators, and blockchain-analysis services recognize transactions involving a designated person or blocked property.

The address functions similarly to an account number, but the transparency of many public blockchains creates additional possibilities. Anyone with access to the ledger can inspect transactions involving the address. Analysts may trace flows among addresses, identify clusters, observe attempts at obfuscation, and follow funds through multiple transactions.

This apparent transparency must be interpreted cautiously. An address does not necessarily reveal the legal identity of the person controlling it. One person may control many addresses. Several people or an organization may share control of one address. An exchange may hold assets for thousands of customers through a limited set of addresses. A smart contract may receive funds from anyone without possessing a conventional human owner.

Address attribution is therefore an investigative conclusion rather than a fact evident from the characters alone.

A sanctions listing may state that a particular address is associated with a blocked person. It does not imply that every address controlled by that person has been identified. Nor does it establish that every address interacting with the listed address is itself sanctioned.

The difference can be expressed as:

listed address → known or alleged association with a sanctioned actor

unlisted address → not necessarily free of such association

connected address → not automatically owned by the same actor

The public nature of blockchain records encourages automated screening. Companies can compare customer withdrawals and deposits against sanctioned-address databases. They can also analyze proximity to addresses associated with ransomware, theft, terrorist financing, narcotics trafficking, or state-sponsored cyber operations.

Such systems are useful but can create false certainty. A transaction may pass through an address without the current user knowing its full history. Assets are interchangeable and may circulate through many innocent hands. The farther analysis moves from an exact listed address toward indirect association, the more judgment is required.

The sanctioned number is therefore not a complete legal conclusion. It is an identifier embedded within a larger compliance investigation.

10.3 Publishing an Address Versus Transacting With It

A public blockchain address may be reproduced for many legitimate purposes. Journalists publish addresses while reporting on thefts and sanctions. Researchers analyze them to study financial networks. Courts include them in evidence. Compliance officers compare them against customer transactions. Investigators use them to trace assets.

Publication is not the same as dealing in blocked property.

This distinction parallels the treatment of other sensitive numbers. Printing a stolen credit-card number may expose a credential, but discussing the existence of a compromised payment system is not the same as using the card. Publishing a file hash associated with malware is not the same as executing the malware. Reproducing a cryptocurrency address in a sanctions report is not the same as sending funds to it.

The relevant activities should be separated:

identifying the address;

describing its transaction history;

monitoring its activity;

receiving an unsolicited transfer from it;

sending assets to it;

providing services that facilitate its transactions;

controlling the private key associated with it.

These acts differ substantially in function and intent.

The distinction becomes especially important because blockchain transfers can sometimes be sent without the recipient's consent. A person who knows a public address may send a small quantity of a token to it. The recipient cannot necessarily prevent the transaction from appearing on the public ledger.

A hostile actor could therefore attempt to create misleading associations by sending assets from a suspicious address to an innocent address. This practice demonstrates why transactional contact alone cannot always establish coordination or control.

The blockchain records that a transfer occurred. It does not automatically explain why it occurred, whether the recipient requested it, or who controlled the initiating address at the time.

Legal interpretation requires additional evidence.

This produces the same recurring lesson found throughout the history of illegal numbers: exact numerical data can create an illusion of complete objectivity. The ledger may record the address and amount precisely, while leaving human intention unresolved.

10.4 Tornado Cash and the Limits of Property-Based Sanctions

Tornado Cash was designed to increase transactional privacy on Ethereum. Public blockchains ordinarily allow observers to trace transfers from one address to another. A privacy protocol can make it more difficult to connect a deposit with a later withdrawal.

In simplified form, a user deposits a standardized quantity of cryptocurrency into a pool and later presents cryptographic information permitting withdrawal to a different address. When many users participate, an observer may be unable to determine with confidence which withdrawal corresponds to which deposit.

The privacy set becomes more useful as participation increases:

more deposits + standardized amounts + delayed withdrawals = greater difficulty linking sender and recipient

Such systems have legitimate applications. A person may not want every customer, donor, political organization, employer, or criminal observer to reconstruct an entire financial history from one public address.

The same privacy can be used to conceal stolen or illicit funds. Hackers, ransomware operators, sanctions evaders, and money launderers may attempt to break the visible connection between the source and destination of assets.

The United States initially sanctioned Tornado Cash in 2022, identifying the service and numerous associated Ethereum addresses. The action reflected concerns that the protocol had been used to launder large quantities of stolen cryptocurrency, including assets associated with North Korean cyber operations.

The resulting litigation did not establish that privacy tools are universally lawful or that sanctions cannot reach cryptocurrency activity. The narrower issue concerned whether the immutable smart contracts at issue qualified as property under the statutory authority Treasury had invoked.

The Court of Appeals concluded that they did not. The contracts were not ownable in the conventional sense, could not be altered or removed by the developers, and did not constitute contracts between legally identifiable parties merely because the software was called a smart contract.

The decision illustrates the difficulty of applying property concepts to autonomous code. Traditional sanctions assume an actor who owns or controls an asset. The government blocks the asset by prohibiting covered persons from transferring or dealing in it.

An immutable program may have no owner capable of exercising the ordinary rights associated with property. Its creators may no longer be able to modify, revoke, move, sell, or destroy it.

The court’s reasoning did not declare the code harmless. It addressed the statutory boundary of a specific sanctions authority. Congress could potentially create different legal tools. Governments can also prosecute unlawful conduct involving software without classifying the software itself as blocked property.

Treasury subsequently removed the economic sanctions against Tornado Cash. That delisting did not transform every use of a mixing protocol into lawful conduct. Transactions involving

sanctioned persons, stolen property, money laundering, or other crimes remain subject to separate legal rules.

The distinction is fundamental:

sanctioning a person → restricts dealings with that person and blocked property

sanctioning an address → uses the address as an identifier or locus of relevant assets

sanctioning immutable code → attempts to treat autonomous software as legally blockable property

The Tornado Cash controversy revealed that these categories cannot always be collapsed into one another.

10.5 Code Without an Owner

Conventional software usually has an operator. A company controls the server, developers release updates, administrators manage accounts, and owners can discontinue the service.

A smart contract deployed to a blockchain may be designed differently. Its code is stored or represented within the distributed network. Participants submit transactions that cause the network to execute the contract according to predetermined rules.

Some contracts remain mutable. An administrator may retain the power to update code, pause operations, redirect funds, change parameters, or replace components. These contracts still have a point of control.

Other contracts are made immutable. The administrative authority is removed or never created. Once deployed, the code continues to operate according to its existing logic whenever valid transactions invoke it.

The difference is critical:

mutable code → identifiable party may retain technical control

immutable code → no party may possess the power to alter or terminate execution

Immutability does not mean that the program operates continuously without input. Many smart contracts remain dormant until a user submits a transaction. Nor does it mean that every surrounding component is decentralized. Websites, user interfaces, governance systems, relays, development teams, and communication channels may remain under human control.

The core contract can nevertheless survive the disappearance of those surrounding structures.

This produces the possibility of code without an owner in the ordinary practical sense. The original developer may have authored it but no longer control it. Token holders may influence related projects without possessing authority over the immutable program. Users interact with the program but do not own its code as a functioning network object.

Traditional legal categories strain under these conditions. An object normally has someone who can possess, transfer, destroy, exclude others from, or exercise control over it. Immutable blockchain code may be publicly accessible and incapable of exclusive possession.

It resembles a published mathematical rule embedded in infrastructure.

Yet the program can still affect property. It may receive deposits, verify proofs, transfer assets, exchange tokens, distribute payments, or enforce conditions.

The resulting combination is historically novel:

ownerless execution + property consequences

Earlier illegal numbers generally depended on an institution recognizing them. A password worked because a server accepted it. A product key worked because licensing software honored it. An autonomous smart contract can continue operating because a distributed network executes its code, even when no administrator wishes to accept responsibility for the result.

10.6 Smart-Contract Addresses as Entrances to Processes

An ordinary cryptocurrency address often identifies a destination associated with a key. A smart-contract address identifies deployed code that can receive inputs and perform operations.

The visible address therefore functions as an entrance to a process.

A user sends a transaction containing the contract address, the requested operation, and any necessary values or data. Network participants validate the transaction and execute the contract according to protocol rules.

The relationship can be represented as:

contract address + valid input + network execution = programmed result

The address does not contain the entire source code in the same way that the DeCSS prime encoded recoverable software. It identifies where the blockchain's executable state can be invoked.

This makes the address simultaneously descriptive and operational.

It identifies a program.

It routes an instruction.

It may receive assets.

It may trigger computation.

It may generate a result that affects other addresses.

The legal significance of a smart-contract address therefore differs from that of a conventional bank-account number. A bank-account number points toward an account managed by an institution. A smart-contract address may point toward an automated mechanism.

This distinction also complicates the meaning of “sending money to an address.” Assets sent to a personal address may come under the control of a key holder. Assets sent to a smart contract may become subject to programmed rules. No person may have discretionary authority to return them.

An address can consequently operate like a machine location. Knowing the address allows users to invoke a process, just as knowing a network address and protocol may allow access to a digital service.

The number is not merely where something is. It is how one reaches something that acts.

10.7 Decentralization and the Failure of Traditional Seizure

Law developed around objects and organizations that could be located and controlled. A government can seize a printing press, close a factory, freeze a bank account, revoke a corporate license, confiscate a server, or order a service provider to stop operating.

Distributed blockchain systems weaken some of these remedies. The ledger may be replicated across many countries. Validators may be operated by unrelated parties. The code may be publicly available. Users may interact through several interfaces or directly through network software.

Authorities can still exert substantial pressure. They may:

close centralized websites;

seize domain names;

sanction identifiable persons;

prosecute developers or operators for unlawful conduct;

require regulated exchanges to block transactions;

restrict access through commercial infrastructure;

seize assets when private keys or custodians are located;

discourage users through legal and financial risk.

These actions can reduce liquidity, usability, legitimacy, and public access. They may make a protocol difficult for ordinary users to reach even when the underlying code remains technically available.

Yet none necessarily deletes immutable code from the blockchain.

This creates a separation between technical survival and practical accessibility. A program may continue to exist while losing its website, development team, institutional support, and connection to regulated markets. It may be unstoppable in one sense but highly suppressible in another.

Decentralization is therefore not absolute immunity from law. It changes where enforcement pressure can be applied.

Instead of regulating the central machine, authorities regulate the edges:

the people who create tools;

the interfaces through which users reach them;

the exchanges where assets enter and leave;

the custodians who hold keys;

the businesses that provide infrastructure;

the persons who initiate transactions.

This strategy resembles regulation of information more broadly. A mathematical relationship cannot be seized, but teachers, publishers, laboratories, platforms, and applications can be controlled.

The core information survives. Its social usefulness may not.

10.8 The Legal Problem of Unstoppable Execution

Autonomous code creates a problem that traditional legal orders were not designed to solve. Law normally issues commands to persons:

stop operating the service;

return the property;

freeze the account;

remove the content;

destroy the device;

compensate the injured party.

An immutable program cannot read a court order, form an intention, appear before a judge, or alter its own code in response to legal judgment.

The law must therefore identify human actors around the program. That may include developers, deployers, administrators, governance participants, interface operators, relayers, users, exchanges, or asset custodians.

Responsibility becomes difficult when these roles are separated. One person may write the code. Another deploys it. Thousands use it. A decentralized network executes it. Token holders vote on optional features. Independent websites provide interfaces. No participant controls the whole system.

Causation becomes distributed.

The challenge can be stated as a sequence of questions:

Who caused the program to exist?

Who caused the disputed transaction?

Who had the power to prevent it?

Who benefited from it?

Who knew of the unlawful purpose?

Who possessed legal authority over the relevant property?

Who can provide an effective remedy?

Different cases may produce different answers. A developer who writes general-purpose code is not necessarily responsible for every later misuse. A person who deliberately deploys a system to facilitate crime presents a different question. An interface operator who knowingly serves sanctioned users may occupy another position. A user who initiates an unlawful transaction may remain directly responsible even when the contract executes automatically.

The danger lies in choosing between two extreme positions.

The first extreme treats code as entirely neutral and assumes that no human bears responsibility once a program is deployed.

The second treats every contributor to a technology as responsible for every harmful use.

Neither position is adequate. The first allows deliberate design for unlawful purposes to hide behind automation. The second threatens legitimate programming, privacy tools, open-source development, and general technological research.

A defensible approach must examine control, knowledge, intention, material assistance, and proximity to the unlawful act.

Autonomous execution also creates remedial problems. A court may determine that a transaction was unlawful without possessing the technical power to reverse it. Assets may have passed through many addresses or been exchanged across several systems. A contract may continue accepting transactions after its use has been prohibited.

Law then becomes partly prospective and indirect. It deters users, regulates intermediaries, imposes liability, and blocks conversion into conventional financial systems. It may not be able to undo the protocol's historical record.

This represents another major stage in the history of illegal numbers.

A Roman death date could coordinate a conspiracy, but humans still had to act.

A 2600-Hz tone caused a telephone switch to respond, but the telephone company could redesign the network.

A leaked password opened an account, but the administrator could revoke it.

A product key activated software, but the issuer could blacklist it.

An immutable smart-contract address may invoke a process that no original issuer can revoke.

The architecture has removed the central authority that once determined whether a number continued to work.

This does not make the number itself illegal. The address may be printed, studied, archived, and analyzed. Its legal significance emerges when someone uses it to initiate, facilitate, conceal, or complete a prohibited transaction.

The governing formula remains relational:

address + code + network + user input + legal context = consequence

The cryptocurrency era therefore deepens the central argument of this paper. Numbers do not become dangerous merely by representing information. They become dangerous when social and technical systems assign them causal power.

A public address can identify blocked property.

A private key can transfer it.

A smart-contract address can activate autonomous code.

A sanctions list can alter how institutions treat the transaction.

A court can prohibit human participation while remaining unable to erase the underlying process.

The resulting struggle is no longer simply between secrecy and disclosure. It is between legal authority and technical persistence.

The illegal number has evolved from a forbidden symbol into an address for a machine that may continue operating after every identifiable owner has stepped away.

11. Exploit Strings, Adversarial AI Tokens, Synthetic DNA, and Information Hazards

The twenty-first century has produced strings whose significance extends beyond symbolism, identification, or access. These strings can exploit software, manipulate artificial-intelligence systems, specify biological structures, or encode methods capable of undermining global security infrastructure.

The common feature is not a particular alphabet. A dangerous string may consist of punctuation, computer commands, model tokens, nucleotide letters, mathematical instructions, or a mixture of symbols. Its importance lies in the relationship between the sequence and a system capable of interpreting it.

Earlier operational numbers generally worked because engineers deliberately designed machines to recognize them. Telephone switches were built to respond to signaling tones. Licensing systems were built to accept product keys. Cryptographic systems were built to recognize valid private keys and signatures.

Modern exploit strings often work for the opposite reason. They trigger behavior that the designers did not intend. A database interprets user data as a command. A parser mishandles a malformed file. A language model gives unexpected weight to an optimized sequence of tokens. A biological design becomes dangerous when combined with synthesis technology and laboratory capability.

This marks a transition from authorized command languages to adversarial command discovery. The dangerous sequence may not have been assigned any official meaning. Its function emerges

from a flaw, statistical vulnerability, unforeseen interaction, or incomplete separation between data and control.

The resulting governance problem is difficult. The same information can support legitimate research, defensive testing, medicine, and technological progress while also facilitating abuse. Regulation must therefore distinguish knowledge from deployment, vulnerability discovery from attack, biological research from harmful construction, and cryptographic analysis from exploitation.

11.1 Malicious Inputs as Operational Strings

Computer systems constantly receive input from users, files, sensors, networks, and other programs. Secure design requires the system to distinguish information that should be treated as data from information that should be treated as instruction.

Many software vulnerabilities arise when that distinction fails.

A database application may expect a user to submit a name, date, or search phrase. If the application inserts that input into a database command without proper separation, the database may interpret part of the user's text as an instruction rather than as ordinary data.

A document reader may expect a file to follow a defined structure. A specially malformed file can trigger an error in memory handling. A network service may expect a packet of limited size or format. An attacker may send an input designed to produce an overflow, crash, unauthorized disclosure, or change in program control.

In each case, the input becomes operational because of the receiving system's vulnerability.

The structure is:

attacker-controlled input + vulnerable interpreter = unintended operation

The string does not carry universal power. It works only in relation to a particular system, implementation, configuration, and state.

An input that compromises one software version may do nothing to another. A patch can remove the vulnerability. A defensive filter can alter the input. A different character encoding can change how the system interprets it.

This conditionality distinguishes exploit strings from ordinary passwords. A password works because the system intentionally accepts it. An exploit works because the system unintentionally accepts an interpretation that its designers failed to prevent.

The exploit may be extremely short relative to its consequences. A small input can expose a database, interrupt a service, or redirect execution. This asymmetry gives malicious strings their

information-hazard character: a few characters may activate capabilities requiring millions of lines of vulnerable software to exist.

The string is therefore not a self-contained weapon. It is a key fitted to an accidental lock.

11.2 Authorization as the Legal Boundary

Exploit strings reveal why technical capability and legal permission must be separated.

A security researcher may send unusual inputs to a system in a controlled laboratory to determine whether a vulnerability exists. A software company may authorize penetration testing of its own products. A university may study exploit behavior in an isolated environment. A defensive team may reproduce an attack to develop a patch.

An attacker may use the identical input against a hospital, bank, government office, industrial system, or private account without permission.

The characters do not change. Authorization does.

The legal significance can be summarized as:

exploit + authorized test environment = security research

exploit + owner-approved assessment = defensive testing

exploit + unauthorized target = potential computer intrusion

exploit + intent to damage or steal = aggravated misuse

This boundary is not always simple. Researchers may discover vulnerabilities while interacting with publicly accessible systems. Terms of service may be unclear. A company may welcome reports in principle but dispute the methods used to obtain them. Testing may begin harmlessly but expose sensitive information belonging to third parties.

Responsible vulnerability-disclosure programs attempt to reduce this uncertainty. They define permitted targets, acceptable methods, reporting channels, confidentiality expectations, and limits on data access.

Such programs recognize that suppressing knowledge of vulnerabilities does not make systems secure. Unknown weaknesses remain available for independent discovery by malicious actors. Defensive progress often requires someone to demonstrate that the flaw is real.

The central ethical question is therefore not whether exploit information exists. It is how discovery, verification, publication, and remediation are managed.

Premature release of a fully operational attack may expose unpatched systems. Excessive secrecy may leave users unaware of danger and allow vendors to postpone repairs. The appropriate balance depends on the severity of the flaw, availability of a patch, prevalence of vulnerable systems, and likelihood of independent exploitation.

Exploit strings are consequently information hazards whose risk changes over time. Before a patch, publication may create immediate danger. After widespread repair, the same information may serve primarily as historical and educational material.

11.3 Adversarial AI Suffixes

Artificial-intelligence systems introduce a novel form of operational string. A language model receives text as sequences of tokens and predicts subsequent tokens according to patterns learned during training and modified through later alignment procedures.

Developers attempt to prevent models from producing certain harmful outputs. They may use training examples, preference optimization, system instructions, classifiers, monitoring, and external safeguards.

These protections are imperfect. Users have developed jailbreak prompts designed to cause a model to disregard or circumvent restrictions. Some jailbreaks rely on role-playing, fictional framing, encoded instructions, emotional manipulation, or elaborate conversational setups.

Adversarial suffixes differ from ordinary hand-written jailbreaks. They can be found through algorithmic optimization. A search procedure explores token combinations and identifies sequences that increase the probability that a model will comply with requests it would otherwise reject.

The resulting suffix may look like grammatical debris. It may contain fragments of words, punctuation, unusual capitalization, or combinations that carry no coherent meaning to a human reader.

Its effect arises inside the model.

The basic structure is:

restricted request + optimized token suffix → increased probability of compliance

The suffix does not necessarily command the model in an ordinary linguistic sense. It perturbs the probability distribution over possible responses. It may increase the likelihood of affirmative language, weaken the effect of refusal patterns, or push internal representations toward regions associated with compliance.

This is an important departure from conventional access codes. No engineer deliberately selected the suffix and programmed the model to obey it. The string is discovered by searching the learned system.

It is therefore better understood as an adversarial stimulus than as a password.

11.4 Machine Meaning Without Human Meaning

Human language is interpreted through grammar, semantics, context, and social knowledge. Machine-learning systems process language through numerical representations learned from vast datasets.

A token has effects determined not only by its dictionary meaning but by its statistical relationships within the model. Sequences that appear meaningless to people may influence internal activations in consistent ways.

This produces machine meaning without clear human meaning.

A person looking at an adversarial suffix may see random characters. The model processes a structured sequence of token identifiers associated with learned vectors and conditional probabilities.

The distinction resembles earlier machine-control systems but is more opaque.

A telephone engineer could explain why a switch responded to 2600 Hz. The signal had an assigned function within the network protocol.

A software engineer can often explain why an injection string works. The parser treats a particular character as the beginning or end of a command.

An AI researcher may identify that a token sequence changes output probabilities without being able to give a complete human-readable explanation for every internal step.

The model's response emerges from billions of interacting parameters. The suffix's function is distributed across that learned structure.

This opacity complicates regulation. A prohibited list of known strings may block previously discovered suffixes while missing countless variants. Small changes may preserve or destroy effectiveness. New models may respond differently. Attackers can search for additional triggers faster than human reviewers can catalog them.

The problem is not that a particular gibberish string has become permanently illegal. The deeper problem is that optimization can repeatedly discover new strings that exploit the model's statistical organization.

11.5 Model Dependence and Ephemeral Functionality

Adversarial AI strings are often dependent on the model and environment in which they are used.

Their effectiveness may vary with:

the model family;

the exact model version;

the tokenizer;

the system prompt;

the preceding conversation;

the sampling settings;

the alignment method;

external safety classifiers;

software updates;

the user’s underlying request.

A string that affects one open-source model may fail against another. A suffix that works today may stop working after an update. A sequence optimized for one class of requests may not generalize to another.

Some research has nevertheless shown transferability. A suffix developed using one or more models may influence models not directly used during optimization. This suggests that different systems can share exploitable structural tendencies.

Transferability makes adversarial suffixes more concerning because an attacker may optimize against an accessible model and then test the result against a closed system.

Even so, no short sequence should be treated as a universal master key for all artificial intelligence. Claims that one mysterious code permanently disables every model’s safeguards misunderstand the technology.

Adversarial strings are typically contingent artifacts. Their function exists within a moving technical ecosystem.

This ephemerality has legal implications. A published string may be operationally potent for a brief period and later become inert. A historical record of the string may remain, but the model that gave it power may no longer exist.

The legal status of such information should therefore depend more on use, authorization, purpose, and resulting harm than on permanent blacklisting of the characters.

11.6 DNA as Executable Biological Information

Synthetic biology extends the concept of executable information from machines into living systems.

DNA is commonly represented through sequences of four letters:

A

C

G

T

These letters correspond to nucleotide bases. Within cells, sequences participate in processes that produce RNA, proteins, regulatory effects, and inherited biological characteristics.

A digital DNA sequence can be stored, copied, transmitted, searched, edited, synthesized, and assembled. The sequence may move from a database into physical material through a synthesis provider or laboratory instrument.

This creates a striking informational chain:

digital sequence → synthesized nucleic acid → biological system → physical effect

The word executable must be used carefully. DNA does not behave like a standalone computer program. Its effects depend on cellular machinery, organism, genomic location, regulatory context, environmental conditions, and interactions with other sequences.

A sequence that performs one function in one organism may behave differently in another. A short fragment may be harmless by itself. A complete biological capability may require numerous components assembled in the correct arrangement.

Nevertheless, DNA is operational information in a profound sense. A symbolic sequence can guide the construction of physical molecules that participate in living processes.

This collapses a boundary that remained intact in earlier examples. A DeCSS prime reconstructed software. A cryptographic key changed access to digital media. A DNA design can contribute to the production of matter.

The resulting information hazards are potentially severe. Some biological sequences relate to toxins, virulence factors, immune evasion, host range, or other functions that could increase risk if misused.

At the same time, biological sequence information supports medicine, diagnostics, vaccine development, agriculture, evolutionary research, and public-health surveillance. A policy of generalized suppression would obstruct essential science without eliminating the underlying biological facts.

11.7 Sequence Screening and Synthetic Biology Governance

Nucleic-acid synthesis providers can screen orders before manufacturing them. Screening commonly involves two related questions.

First, who is the customer?

Second, what is being ordered?

Customer screening attempts to verify identity, institutional affiliation, legitimacy, and intended use. Sequence screening compares requested material with information associated with biological risks or regulated agents.

The objective is not to declare certain combinations of A, C, G, and T universally illegal. It is to identify orders that warrant additional review, clarification, refusal, or referral.

A risk assessment may consider:

the biological function associated with the sequence;

the length and completeness of the requested material;

whether fragments can be assembled;

the customer’s identity and institutional context;

the type of synthesis equipment involved;

the intended research purpose;

the availability of appropriate containment and oversight.

This approach recognizes that sequence risk is relational. A fragment may have little significance alone but greater significance when ordered with complementary components. A legitimate

public-health laboratory and an unverifiable purchaser may request similar material for very different purposes.

Screening is technically difficult. Biological evolution produces variation. Harmful functions may not be represented by one exact sequence. Benign organisms may contain regions resembling sequences found in dangerous organisms. Attackers may divide orders among providers or request modified fragments.

A purely literal blacklist is therefore inadequate. Screening must account for similarity, function, assembly, and context while limiting false positives that burden legitimate science.

The governance challenge resembles sanctions screening and malware detection. Exact matching is efficient but narrow. Broader similarity analysis captures more risk but produces more ambiguity.

No screening system can eliminate all danger. Its value lies in raising barriers, generating scrutiny, and making suspicious acquisition more difficult.

11.8 Fragments, Assembly, and the Missing Decoder

Dangerous information is often distributed across components that appear harmless separately.

A software exploit may require a vulnerable program, a particular input, and a specific system configuration.

An adversarial AI suffix may require a target model and a harmful underlying prompt.

A biological capability may require multiple sequence fragments, regulatory elements, assembly methods, laboratory equipment, and a suitable host system.

A cryptographic attack may require an algorithm, captured data, computing resources, and implementation knowledge.

The general pattern is:

component A + component B + environment E + procedure P → operational capability

This creates the missing-decoder problem. A sequence may not be dangerous in isolation because its significance becomes apparent only when combined with external knowledge or infrastructure.

In the DeCSS prime, a decoder transformed decimal digits back into software. In synthetic biology, the decoder is partly biological and partly technical: synthesis machinery, cells, enzymes, laboratory procedures, and scientific knowledge.

The law often seeks identifiable objects. Distributed capability resists this model. No individual fragment may satisfy a statutory definition. No participant may possess the entire system. Different providers may unknowingly supply separate components.

Fragmentation can be intentional. A person may divide information or materials to avoid detection. It can also arise naturally from scientific specialization and supply chains.

Effective governance must therefore consider aggregation without treating every common fragment as suspicious. Many sequences and tools used in hazardous contexts also have widespread legitimate applications.

The danger lies in combination, but overbroad regulation of components can paralyze ordinary research.

11.9 Algorithms That Could Collapse Cryptographic Security

An algorithm capable of defeating widely deployed cryptography would be another profound information hazard.

Modern communications depend on cryptographic systems for confidential messages, financial transactions, software updates, identity verification, government records, medical data, and critical infrastructure.

Some public-key systems rely on mathematical problems believed to be computationally difficult for conventional computers. A sufficiently powerful new algorithm, implementation breakthrough, or cryptographically relevant quantum computer could undermine those assumptions.

The danger would not be limited to future messages. Adversaries can collect encrypted information today and retain it in the hope that later capabilities will permit decryption.

An effective cryptographic attack could be represented in several forms:

a mathematical theorem;

a research paper;

source code;

optimized software;

specialized hardware design;

a remote service;

an autonomous attack system.

As in the earlier encryption-export disputes, the distinction between idea and capability would become unstable. A proof might enable implementations. Source code could spread globally in minutes. Once independently verified and replicated, suppression would be unlikely to restore security.

The appropriate response would be migration rather than censorship. Vulnerable algorithms would need to be replaced, keys regenerated, protocols updated, and systems redesigned.

This is why cryptographic agility matters. A secure infrastructure should be capable of changing algorithms without complete reconstruction.

The lesson resembles the response to leaked product keys and telephone signaling flaws. Once a foundational assumption fails, attempting to suppress the revealing number or algorithm is insufficient. The architecture must change.

11.10 Distributed Knowledge and Threshold Possession

Modern information hazards can be divided among people, organizations, files, or systems.

Secret-sharing methods can divide a cryptographic key into several fragments so that no single fragment confers control. A specified number of fragments must be combined before the key can be reconstructed.

Scientific projects similarly distribute expertise. One participant understands sequence design. Another operates synthesis equipment. Another has access to a laboratory. Another controls computational analysis. No individual possesses the complete capability alone.

This creates threshold possession.

The operational capability exists at the level of the group even though no individual holds every component.

A simple model is:

share 1 + share 2 + share 3 → capability

share 1 alone → insufficient

The legal system must determine when coordination converts partial possession into collective responsibility.

Mere knowledge of one component should not automatically establish participation in a harmful project. Most advanced technologies depend on specialized contributions. Researchers frequently work on parts of systems whose full applications they do not control.

Liability becomes more plausible when participants share unlawful intent, coordinate their contributions, understand the larger objective, and materially assist its completion.

The problem resembles conspiracy law but is intensified by automation and global collaboration. Participants may never meet. Their contributions may occur across different countries and times. Some components may be produced by autonomous tools.

Artificial intelligence further complicates responsibility. A model may generate code, suggest designs, or connect fragments of knowledge that no single human contributor assembled manually.

Distributed knowledge therefore weakens the assumption that dangerous information has one author, possessor, or publisher.

11.11 The Convergence of Code and Matter

The historical progression examined in this paper leads toward a convergence of symbolic information, machine action, and physical consequence.

Ancient numbers expressed cosmological order.

Roman dates influenced political behavior.

Medieval numerals recorded commercial value.

Telephone tones controlled switches.

Cryptographic keys controlled access.

Software exploits redirected computation.

Blockchain addresses invoked autonomous contracts.

AI suffixes manipulated learned models.

DNA sequences guided biological construction.

At each stage, the distance between symbol and consequence decreased.

The modern string can no longer be classified only by what it means to a human reader. It must also be evaluated by what systems can do with it.

This does not mean that every operational sequence should be prohibited. Such a policy would be impossible and destructive. Computer security requires exploit knowledge. AI safety requires adversarial testing. Medicine requires biological sequences. Cryptographic progress requires study of attacks.

The challenge is to govern transitions from information to harmful capability.

A useful framework asks:

How direct is the path from sequence to consequence?

What additional equipment, knowledge, or access is required?

Does the sequence have substantial legitimate uses?

Is the recipient authorized to use it?

Can the relevant system be redesigned to reduce dependence on secrecy?

Would publication create immediate risk, or would it primarily advance defensive understanding?

Can access be limited without suppressing general scientific knowledge?

Are safeguards available at the point of execution rather than at the point of expression?

These questions shift attention away from the fantasy of an intrinsically illegal string. They focus instead on capability, context, and control.

The strongest defenses often operate at the system level.

Software should separate data from commands.

Networks should authenticate control signals.

Credentials should be revocable and narrowly scoped.

AI systems should use layered safeguards rather than depend on blocking known suffixes.

Synthesis providers should combine customer verification with sequence screening.

Cryptographic systems should be designed for algorithm replacement.

Laboratories should maintain appropriate oversight and containment.

These measures address the conditions that give information operational power.

The central lesson is architectural. A society cannot make dangerous knowledge disappear merely by prohibiting one representation. A sequence can be transformed, fragmented, rediscovered, translated, or regenerated by algorithms.

The sustainable response is to build systems that fail safely, limit the consequences of disclosure, authenticate authority, monitor misuse, and preserve legitimate research.

The illegal number has now reached its most consequential form. It is no longer only a forbidden idea, secret credential, or coded political message. It may be an adversarial input discovered by a machine, a digital design translated into biological material, or an algorithm capable of altering the security assumptions of civilization.

The final section will draw these histories together into a general theory. It will ask when a string should be understood as speech, property, evidence, access, instruction, infrastructure, or action—and whether law can regulate its harmful uses without attempting the impossible task of regulating mathematics itself.

12. A General Theory of Illegal Numbers and the Future of Executable Information

The history examined in this paper shows that an "illegal number" is rarely illegal as mathematics. The integer itself does not become unlawful in every place, language, notation, or context. What governments, courts, institutions, corporations, and communities attempt to regulate is the representation, circulation, interpretation, or use of a sequence.

This distinction resolves the apparent paradox. A number may be abstract, but its representation can participate in concrete conduct. A date can coordinate political action. A password can open an account. A private key can transfer assets. A software string can exploit a computer. A model-specific suffix can alter an artificial-intelligence system's behavior. A nucleotide sequence can contribute to the construction of biological material.

The legal meaning of a string therefore does not reside in its characters alone. It emerges from a relationship among the sequence, the interpreter, the system, the holder, the audience, the purpose, and the resulting consequence.

The most important historical change is the shortening distance between information and action. Ancient numerical restrictions focused primarily on meaning. Modern restrictions increasingly focus on functionality. A sacred number offended through interpretation. A cryptographic key acts through execution. A biological sequence may help transform digital information into physical matter.

The future of numerical governance will depend on whether legal systems can distinguish harmful conduct from mathematical existence, dangerous deployment from legitimate research, and operational capability from mere description.

12.1 Numbers Are Rarely Illegal as Mathematical Objects

An integer exists independently of its use. The number 2600 remains the same whether it describes a telephone frequency, a year, a room number, a price, or a quantity of objects. The

number 187 remains an ordinary integer even when a particular community understands it as shorthand for murder. A large prime remains prime whether it is studied for number theory or used to encode software.

This independence means that law cannot alter the mathematical properties of a number. A statute cannot make a prime composite, make √2 rational, or remove zero from positional notation.

Law can only regulate human conduct involving representations of those objects.

A prohibition may apply to publication, possession, transmission, use, trafficking, display, or incorporation into a technical process. The regulation therefore attaches not to the abstract number but to an act involving the number.

This can be stated as:

abstract sequence + no context = mathematically neutral object

abstract sequence + representation + actor + purpose + system = potentially regulated conduct

The distinction matters because public discussions often use the phrase “illegal number” to suggest that a government has declared arithmetic itself unlawful. This is usually inaccurate.

Even the strongest examples involve context. A stolen private key may be unlawful to use but lawful to reproduce in a judicial opinion after it has been invalidated. A prohibited political code may be restricted on a government-issued license plate but protected in a scholarly history. A decryption key may be actionable when distributed as part of a circumvention service yet discussable in legal analysis.

The same sequence can therefore occupy several legal categories at once.

The problem is not that numbers become illegal. The problem is that societies increasingly build systems in which numbers perform legally significant actions.

12.2 A Taxonomy of Restricted Numbers

The phrase “illegal number” covers several distinct categories that should not be confused.

A secret number is protected because an institution or person wishes to preserve confidentiality. A password, unpublished key, or confidential account code may fall into this category. Its secrecy may be contractual, commercial, personal, or strategic.

A restricted number may be lawfully known by authorized persons but not distributed freely. Military codes, certain classified identifiers, or protected authentication information may be examples.

An unauthorized credential is a sequence acquired, possessed, transferred, or used without the permission of the system owner or rightful account holder.

A prohibited-use string is lawful to discuss or possess but unlawful to employ against a protected target. An exploit used in an authorized laboratory differs from the same exploit used to damage a hospital network.

A symbolic code is regulated because of what it communicates to human beings. Political dates, extremist numbers, gang codes, and threatening statutory references belong here.

An evidentiary string is not itself unlawful but helps establish identity, intention, knowledge, association, or participation. A wallet address, file hash, username, or coded phrase may function as evidence.

A contraband-encoding number is a representation from which prohibited material can be reconstructed. The illegal-prime concept belongs to this category.

An autonomous operational string identifies or invokes a process that continues without direct human control, as with some smart contracts.

A biological capability string contributes to the construction or modification of living systems.

These categories may overlap. A private key is both a secret and a capability. A smart-contract address is both an identifier and an operational entry point. A political number may be both symbolic speech and evidence of intent.

A useful legal analysis must therefore begin by asking what type of string is involved. Treating every sensitive sequence as the same kind of object produces confusion and overreach.

12.3 The Equivalence Class of Representations

Every finite digital object can be represented in many forms.

A program may appear as source code, machine code, binary digits, hexadecimal digits, decimal integers, compressed data, colored pixels, musical notes, or an image containing printed text.

These are not necessarily different informational objects. They may be reversible encodings of the same underlying content.

The complete set of reversible forms can be treated as an equivalence class:

[I] = all representations from which information I can be reconstructed

The class may be extremely large. If encryption keys, compression methods, layout choices, image formats, and fragmentation schemes are included, the number of possible representations becomes effectively unbounded.

This creates a deep regulatory problem. A prohibition aimed only at one visible string is easy to evade. A prohibition aimed at the entire equivalence class risks becoming impossibly broad.

Suppose software is prohibited in hexadecimal form. It can be converted to decimal. If decimal is prohibited, it can be embedded in an image. If the image is prohibited, it can be encrypted. If the ciphertext is prohibited, it can be divided across many files or generated from instructions.

The informational identity survives.

This suggests that law should not focus exclusively on surface representation. It should examine whether a person knowingly communicated or used information in a way that substantially enabled prohibited conduct.

Yet this functional approach also creates danger. A sufficiently broad rule could reach books, lectures, mathematical proofs, historical archives, or general scientific knowledge.

The equivalence-class problem therefore exposes the central tension of information law: narrow rules are easy to evade, while broad rules threaten legitimate expression.

12.4 Why Universal Numerical Blacklists Cannot Work

A universal blacklist of dangerous numbers would fail for both technical and conceptual reasons.

First, re-encoding changes visible form. The same information can be represented through different alphabets, bases, file types, languages, and media.

Second, encryption can make identical content appear entirely different. The same file encrypted under two different keys produces unrelated-looking outputs.

Third, fragmentation can divide dangerous information into individually harmless pieces. No single component may trigger the blacklist.

Fourth, context determines function. A known exploit string may be inert against a patched system. A private key may have been revoked. A political code may appear coincidentally.

Fifth, new dangerous strings can be generated faster than lists can be updated. AI adversarial optimization and software exploit discovery can produce previously unknown variants.

Sixth, innocent strings may collide with prohibited patterns. Short numerical codes are especially likely to have unrelated legitimate uses.

A blacklist can therefore serve as one defensive tool but not as a complete solution.

Literal matching works best when the target is stable, exact, and context-independent. It works poorly when the target is mutable, encrypted, approximate, or dependent on surrounding systems.

The danger of blacklists is not only that they miss harmful content. They may also create false confidence. An organization may assume that anything not on the list is safe.

A more resilient approach combines matching with authentication, behavioral monitoring, context, access controls, human review, and system redesign.

The lesson from 2600 Hz applies broadly. The durable solution was not to prohibit every recording of the tone. It was to redesign telephone signaling so ordinary users could no longer issue trusted control commands.

12.5 The Theoretical Limits of Automated Classification

Automatic systems can detect known malware, compare hashes, recognize suspicious patterns, flag extremist codes, screen biological sequences, and identify unusual transactions. These tools are essential at modern scale.

Yet perfect classification is impossible.

A program's behavior may depend on future inputs, hidden state, network responses, dates, hardware conditions, or interactions with other programs. A string that appears harmless in one environment may become dangerous in another.

Theoretical computer science establishes that no general procedure can correctly predict every relevant behavior of every possible program.

Practical systems therefore rely on approximations. They classify by known signatures, statistical similarity, behavioral indicators, reputation, or risk scoring.

These methods produce unavoidable errors.

A strict filter may block innocent material.

A permissive filter may allow dangerous material.

An adaptive attacker can study the filter and design inputs that avoid it.

A continually updated filter may become opaque and difficult to challenge.

The same limitation appears outside conventional computing. Biological screening cannot infer every possible function from sequence alone. Political moderation cannot determine intention from digits alone. AI defenses cannot enumerate every adversarial token combination.

Automated classification is therefore a risk-management process, not an oracle.

The legal system should resist treating a machine-generated flag as complete proof. Automated findings should be reviewable, explainable where possible, and connected to additional evidence.

12.6 False Positives and False Negatives

A false positive occurs when harmless information is classified as dangerous. A false negative occurs when dangerous information passes undetected.

Both errors matter, but institutions often emphasize only one.

Security agencies may focus on false negatives because missing a serious threat can produce catastrophic harm. Civil-liberties advocates may focus on false positives because incorrect classification can suppress speech, block transactions, or stigmatize innocent people.

The correct balance depends on the stakes.

A hospital network may tolerate more false alarms when detecting malware because the cost of a successful attack is high. A public political forum may require greater caution before removing ambiguous symbolic speech. A biological synthesis provider may apply enhanced review to suspicious orders without treating every similarity match as proof of malicious intent.

The most dangerous systems are those that conceal the tradeoff. They present classifications as objective facts while hiding thresholds, uncertainty, and error rates.

A number on a blacklist may appear precise, but the decision to place it there involved human judgment. A sanctions address may be exact, while attribution remains uncertain. A hash match may be exact, while the classification of the underlying file may be wrong. A coded slogan may be recognized correctly, while the speaker's purpose remains misunderstood.

Procedural safeguards are therefore essential. Individuals and organizations need mechanisms to contest classifications, correct errors, and present context.

Precision of representation must not be confused with certainty of interpretation.

12.7 The Essential Variables of Legal Meaning

The legal meaning of a string can be analyzed through a set of variables:

sequence;

representation;

function;

target;

authorization;

intent;

audience;

context;

consequence.

The sequence is the underlying information.

The representation is the form in which it appears.

The function is what the sequence enables or communicates.

The target is the person, system, account, institution, or organism affected.

Authorization asks whether the actor had permission.

Intent concerns what the actor meant or sought to accomplish.

Audience concerns who was expected to interpret or act upon the sequence.

Context includes surrounding conduct, timing, history, and circumstances.

Consequence concerns what actually occurred or could reasonably occur.

These variables explain why identical strings can receive different treatment.

A software exploit sent to a test server with permission is research. Sent to a bank without permission, it may be intrusion.

A political number in a history book is documentation. The same number directed at a person alongside evidence of planned violence may become part of a threat.

A biological sequence in a public database may support medicine. The same sequence ordered in suspicious combinations by an unverifiable purchaser may require intervention.

A private key printed after revocation is historical information. The active key used to transfer stolen assets is an instrument of theft.

No single variable is always decisive. Together they provide a more reliable framework than literal string prohibition.

12.8 Regulating Conduct Rather Than Integers

The strongest general principle emerging from this history is that law should regulate harmful conduct rather than abstract integers.

This does not mean that publication can never be regulated. Distribution may itself be part of harmful conduct when it is intentionally integrated into unauthorized access, trafficking, threats, or prohibited transactions.

The distinction is one of focus.

A conduct-centered rule asks:

Was protected access defeated?

Was property transferred without authorization?

Was a serious threat communicated?

Was harmful code deployed?

Was assistance knowingly provided to prohibited actors?

Was dangerous material constructed or acquired?

A number-centered rule asks only:

Did the sequence appear?

The first approach is more demanding but more accurate. It requires proof of function, context, and responsibility.

The second approach is easier to automate but risks absurdity. It may prohibit scholarship, journalism, mathematics, or accidental matches while failing to stop transformed versions of the same information.

The preference for conduct-centered regulation also encourages better security architecture. Instead of trying to suppress every leaked credential, systems can revoke it. Instead of banning known exploit strings, software can be patched. Instead of censoring every AI suffix, models can be strengthened through layered defenses.

Law remains necessary, but it should complement technical resilience rather than substitute for it.

12.9 Protecting Research, Journalism, Mathematics, and Historical Memory

Any system regulating dangerous information must preserve legitimate inquiry.

Mathematics depends on the freedom to examine abstract relationships.

Computer security depends on the ability to analyze vulnerabilities.

Journalism depends on the ability to document misconduct, sanctions, leaks, and political censorship.

Historical scholarship depends on preserving evidence that governments or institutions may prefer to erase.

Medicine and biology depend on access to sequence data and experimental methods.

Cryptography depends on open testing of algorithms.

The public interest often requires publication of information that is uncomfortable or potentially dual-use.

Blanket suppression creates several harms. It may prevent researchers from identifying weaknesses. It may allow institutions to hide failures. It may obstruct independent verification. It may erase historical memory. It may concentrate knowledge in governments or corporations without public accountability.

Protection does not require unrestricted publication of every operational detail at every moment. Responsible disclosure, temporary access controls, redaction, staged publication, and secure research environments may sometimes be justified.

The governing presumption, however, should remain that knowledge is protected unless a specific and demonstrable risk justifies a narrowly tailored restriction.

A society that treats all powerful knowledge as contraband will weaken the very institutions needed to understand and manage danger.

12.10 From Symbolic Meaning to Causal Power

The historical progression of illegal numbers can be understood as a movement from symbolic meaning toward causal power.

The Pythagorean irrational threatened a worldview.

The Hebrew forms of fifteen and sixteen risked reproducing sacred language.

The Roman imperial death date threatened political stability.

Zero and positional numerals affected documentary trust.

The 2600-Hz tone controlled telecommunications machinery.

Source code became both explanation and executable method.

Decryption keys unlocked protected media.

Credentials conferred account authority.

Blockchain addresses invoked financial processes.

AI adversarial strings altered model behavior.

DNA sequences contributed to biological construction.

The progression is not perfectly linear. Symbolic and operational meanings continue to coexist. Political codes remain powerful, and sacred representations remain significant.

Yet the general direction is clear. Information increasingly acts through systems designed to interpret it.

The number is no longer merely read. It is executed.

This transformation changes legal reasoning. Symbolic speech requires interpretation of meaning. Executable information requires analysis of technical effect. Modern cases often require both.

A smart-contract address may communicate political ideals about decentralization while also moving assets. An AI prompt may be research speech while also inducing harmful output. Source code may explain a vulnerability while also reproducing it.

The same string can be language and mechanism.

12.11 When Speech Becomes Infrastructure

Modern society is built from instructions.

Software routes communications, manages power grids, controls vehicles, processes financial transactions, administers medical equipment, and governs access to information. Code is therefore no longer merely an object within society. It is part of society's infrastructure.

When speech becomes infrastructure, errors and attacks acquire physical consequences.

A sentence in a book influences readers indirectly.

A command in an industrial system may open a valve.

A transaction signed by a private key may transfer wealth.

A sequence submitted to a synthesis platform may initiate production of biological material.

A smart contract may execute financial rules automatically.

This does not eliminate the expressive character of code. It intensifies the need for nuanced treatment.

Infrastructure speech requires both constitutional protection and technical responsibility. Programmers must be able to discuss systems openly, but deployed systems must be held to standards of security and accountability.

The key distinction may be between expression about a system and authorized control of a system. Yet even this boundary can blur when publication directly enables execution.

The law will increasingly confront acts that are simultaneously communication and operation.

12.12 The Future of Numerical Governance

Future numerical governance will face several emerging challenges.

Artificial intelligence may generate exploit strings, encryption attacks, biological designs, and new encodings automatically. The number of potentially dangerous representations may therefore grow faster than human institutions can catalog them.

Autonomous agents may act on credentials without immediate human direction. Determining who intended or authorized the action will become more difficult.

Quantum computing may weaken cryptographic assumptions that protect current communications and assets.

Synthetic biology may further reduce the distance between digital design and biological construction.

Distributed systems may create programs that no person can modify, revoke, or terminate.

Cross-border networks will continue to weaken national control over information.

These developments will make prohibition by enumeration increasingly ineffective.

The future will require systems designed around resilience rather than secrecy alone.

Credentials should be revocable.

Privileges should be limited.

Critical systems should require multiple forms of authorization.

Software should separate data from command.

AI systems should be monitored at several layers.

Biological synthesis should combine sequence screening with customer verification and institutional oversight.

Cryptographic systems should be capable of migration.

Autonomous systems should preserve mechanisms for accountability where feasible.

Legal standards should focus on control, intent, authorization, material assistance, and foreseeable consequence.

The final principle is simple but demanding:

The more power society gives to strings, the more carefully it must design the systems that obey them.

Numbers themselves do not threaten civilization. Fragile institutions, poorly designed machines, concentrated credentials, unaccountable automation, and reckless use give numbers their dangerous force.

The phrase “illegal number” will therefore remain rhetorically useful but conceptually incomplete. It captures the surprise that something as abstract as a sequence can become entangled with crime, censorship, property, warfare, and biological risk.

The deeper truth is that civilization has transformed numbers into instruments.

They identify, authorize, predict, unlock, execute, transfer, persuade, and construct.

The legal challenge is not to outlaw mathematics. It is to govern the expanding causal power that societies have chosen to place inside symbolic form.

13. References

Web-based sources were last accessed June 16, 2026.

13.1 Ancient Mathematics, Pythagoreanism, Sacred Numbers, and Hebrew Numerical Notation

Aristotle. 1984. “Prior Analytics.” In The Complete Works of Aristotle, edited by Jonathan Barnes. Princeton, NJ: Princeton University Press.

Burkert, Walter. 1972. Lore and Science in Ancient Pythagoreanism. Translated by Edwin L. Minar Jr. Cambridge, MA: Harvard University Press.

Euclid. The Elements, Book X. Translated and presented by David E. Joyce. Clark University.

Heath, Thomas L. 1921. A History of Greek Mathematics. Vol. 1, From Thales to Euclid. Oxford: Clarendon Press. Internet Archive.

Huffman, Carl. 2019. “Pythagoreanism.” Stanford Encyclopedia of Philosophy. Stanford University.

Iamblichus. 1991. On the Pythagorean Way of Life. Translated by John Dillon and Jackson Hershbell. Atlanta: Scholars Press.

Knorr, Wilbur Richard. 1975. The Evolution of the Euclidean Elements: A Study of the Theory of Incommensurable Magnitudes and Its Significance for Early Greek Geometry. Dordrecht: D. Reidel.

Kline, Morris. 1972. Mathematical Thought from Ancient to Modern Times. New York: Oxford University Press.

Maimonides, Moses. Mishneh Torah, Foundations of the Torah 6:2. Sefaria.

Scholem, Gershom, and M. Seligsohn. 1906. “Names of God.” The Jewish Encyclopedia. Jewish Encyclopedia.

Shulchan Arukh. Yoreh De’ah 276:9. Sefaria.

Smith, David Eugene. 1958. History of Mathematics. Vol. 2, Special Topics of Elementary Mathematics. New York: Dover Publications.

13.2 Roman Astrology, Imperial Power, and Treason

Barton, Tamsyn. 1994. Ancient Astrology. London and New York: Routledge. Routledge.

Bauman, Richard A. 1974. Impietas in Principem: A Study of Treason Against the Roman Emperor with Special Reference to the First Century A.D. Munich: C. H. Beck. Google Books.

Barton, Tamsyn. 1994. Power and Knowledge: Astrology, Physiognomics, and Medicine Under the Roman Empire. Ann Arbor: University of Michigan Press.

Cramer, Frederick H. 1954. Astrology in Roman Law and Politics. Memoirs of the American Philosophical Society, vol. 37. Philadelphia: American Philosophical Society.

Potter, David. 1994. Prophets and Emperors: Human and Divine Authority from Augustus to Theodosius. Cambridge, MA: Harvard University Press.

Rutledge, Steven H. 2001. Imperial Inquisitions: Prosecutors and Informants from Tiberius to Domitian. London: Routledge.

Suetonius. The Lives of the Twelve Caesars, "Domitian." Perseus Digital Library.

Tacitus. The Annals. Translated by Alfred John Church and William Jackson Brodribb. Perseus Digital Library.

13.3 Hindu-Arabic Numerals, Zero, Fibonacci, and Commercial Recordkeeping

Danna, Raffaele. 2021. "Figuring Out: The Spread of Hindu-Arabic Numerals in the European Tradition of Practical Mathematics." Nuncius 36: 5–48.

Danna, Raffaele. 2023. "The Spread of Hindu-Arabic Numerals Among Practitioners in Italy and England, Thirteenth–Sixteenth Centuries: Two Moments of a European Innovation Cycle?" In Before and After the Commercial Revolution. Florence: Firenze University Press. DOI: 10.36253/979-12-215-0092-9.06.

Fibonacci, Leonardo of Pisa. 2002. Fibonacci's Liber Abaci: A Translation into Modern English of Leonardo Pisano's Book of Calculation. Translated by L. E. Sigler. New York: Springer.

Ifrah, Georges. 2000. The Universal History of Numbers: From Prehistory to the Invention of the Computer. Translated by David Bellos, E. F. Harding, Sophie Wood, and Ian Monk. New York: John Wiley and Sons.

Kaplan, Robert. 1999. The Nothing That Is: A Natural History of Zero. New York: Oxford University Press.

Menninger, Karl. 1969. Number Words and Number Symbols: A Cultural History of Numbers. Translated by Paul Broneer. Cambridge, MA: MIT Press.

Smith, David Eugene, and Louis Charles Karpinski. 1911. The Hindu-Arabic Numerals. Boston: Ginn and Company. Project Gutenberg.

Swiss National Museum. 2020. "How the Number 7 Came to Europe." Swiss National Museum Blog.

13.4 Telegraph Codes, Telephone Signaling, and the 2600-Hz Era

Lapsley, Phil. 2013. Exploding the Phone: The Untold Story of the Teenagers and Outlaws Who Hacked Ma Bell. New York: Grove Press.

Rosenbaum, Ron. 1971. “Secrets of the Little Blue Box.” Esquire, October 1971.

United States v. Brady, 820 F. Supp. 1346, 1355–56, U.S. District Court for the District of Utah, 1993. Justia.

United States v. Foster, 580 F.2d 388, U.S. Court of Appeals for the Tenth Circuit, 1978.

United States v. McNutt, 908 F.2d 561, U.S. Court of Appeals for the Tenth Circuit, 1990.

13.5 Cryptographic Export Controls, Source Code, and Functional Speech

Bernstein v. United States Department of State, 922 F. Supp. 1426, U.S. District Court for the Northern District of California, 1996. Justia.

Bernstein v. United States Department of State, 945 F. Supp. 1279, U.S. District Court for the Northern District of California, 1996. Justia.

Bernstein v. United States Department of State, 974 F. Supp. 1288, U.S. District Court for the Northern District of California, 1997. Justia.

Bernstein v. United States Department of Justice, 176 F.3d 1132, U.S. Court of Appeals for the Ninth Circuit, 1999. Justia.

Bambauer, Derek E. 2014. “Is Data Speech?” Stanford Law Review 66: 57–119. Stanford Law Review.

Junger v. Daley, 209 F.3d 481, U.S. Court of Appeals for the Sixth Circuit, 2000. Justia.

Karn v. United States Department of State, 925 F. Supp. 1, U.S. District Court for the District of Columbia, 1996. Justia.

Karn, Philip R. Jr. 1997. “Prepared Statement on Encryption Export Controls.” United States Congress. National Security Archive.

Lessig, Lawrence. 2006. Code: Version 2.0. New York: Basic Books. Codev2.

Solum, Lawrence B. 2001. “Freedom of Communicative Action: A Theory of the First Amendment Freedom of Speech.” Northwestern University Law Review 83: 54–135.

13.6 The DMCA, DeCSS, Illegal Primes, AACS, and Software Keys

Carmody, Phil. 2001. “The First Illegal Prime.” Prime Curios. The Prime Pages.

Electronic Frontier Foundation. “DVD Copy Control Association v. Bunner and DVD Copy Control Association v. Pavlovich.” EFF Case Archive.

Felten, Edward W. 2007. “AACS: A Tale of Three Keys.” Center for Information Technology Policy, Princeton University. CITP.

Felten, Edward W. 2007. “Digg Users Revolt over AACS Key.” Center for Information Technology Policy, Princeton University. CITP.

Felten, Edward W. 2007. “Why the 09ers Are So Upset.” Center for Information Technology Policy, Princeton University. CITP.

Sony Computer Entertainment America LLC v. Hotz et al., No. 3:11-cv-00167, Document 127, U.S. District Court for the Northern District of California, April 11, 2011. Justia.

Sony Computer Entertainment America and George Hotz. 2011. “Settlement in George Hotz Case.” PlayStation Blog.

United States Copyright Office. Universal City Studios, Inc. v. Corley, 273 F.3d 429, U.S. Court of Appeals for the Second Circuit, 2001. Copyright Office PDF.

United States Code. Title 17, Section 1201. “Circumvention of Copyright Protection Systems.” Cornell Legal Information Institute.

Universal City Studios, Inc. v. Corley, 273 F.3d 429, U.S. Court of Appeals for the Second Circuit, 2001. Justia.

Plummer, Dave W. 2025. “The Story of the Windows XP FCKGW Volume Licensing Key.” Referenced in Mark Tyson, “Legendary Microsoft Developer Reveals the True Story Behind the Most Famous Product Activation Key of All Time.” Tom’s Hardware, October 2025. Tom’s Hardware.

13.7 Political Memory, Protest Codes, Extremist Symbols, and Threat Law

Amnesty International. “The 1989 Tiananmen Crackdown.” Amnesty International.

Anti-Defamation League. “1488.” Hate Symbols Database. ADL.

Anti-Defamation League. “ACAB.” Hate Symbols Database. ADL.

Austrian Constitutional Court. 2019. Decision E 5004/2018, concerning an ACAB banner and freedom of expression. Official English Summary.

Berlin Senate. 2022. “Banned License Plates Still in Circulation in Berlin.” Berlin.de.

Bundesverfassungsgericht. 2015. Order of February 26, 2015, 1 BvR 1036/14, concerning the “FCK CPS” inscription and insult law. Federal Constitutional Court.

California Legislative Information. Penal Code Section 187, definition of murder. California Legislature.

California Legislative Information. Penal Code Section 422, criminal threats. California Legislature.

Citizen Lab. 2013. “Censoring a Commemoration: What June 4-Related Search Terms Are Blocked on Weibo Today?” University of Toronto. Citizen Lab.

Counterman v. Colorado, 600 U.S. 66, Supreme Court of the United States, 2023. Supreme Court opinion.

Ng, Jason Q. 2013. Blocked on Weibo: What Gets Suppressed on China’s Version of Twitter and Why. New York: New Press.

United States Department of Justice. 2026. “Federal Grand Jury Indicts Former FBI Director James Comey for Threats to Harm President Trump.” April 28, 2026. Department of Justice.

United States Code. Title 18, Section 871. “Threats Against the President and Successors to the Presidency.” Cornell Legal Information Institute.

United States Code. Title 18, Section 875. “Interstate Communications.” Cornell Legal Information Institute.

Watts v. United States, 394 U.S. 705, Supreme Court of the United States, 1969. Justia.

Virginia v. Black, 538 U.S. 343, Supreme Court of the United States, 2003. Justia.

13.8 Access Devices, Passwords, Hashes, and Computer Authorization

National Institute of Standards and Technology. 2015. Secure Hash Standard. Federal Information Processing Standards Publication 180-4. NIST.

National Institute of Standards and Technology. 2022. Digital Investigation Techniques: A NIST Scientific Foundation Review. NIST IR 8354. NIST PDF.

National Institute of Justice. 2022. “Digital Caseload Processing with the NIST National Software Reference Library.” NIJ.

United States Code. Title 18, Section 1029. “Fraud and Related Activity in Connection with Access Devices.” Cornell Legal Information Institute.

United States Code. Title 18, Section 1030. “Fraud and Related Activity in Connection with Computers.” Office of the Law Revision Counsel.

United States Department of Justice. “Computer Fraud and Abuse Act.” Justice Manual, Section 9-48.000. Department of Justice.

Van Buren v. United States, 593 U.S. 374, Supreme Court of the United States, 2021. Supreme Court opinion.

13.9 Cryptocurrency Addresses, Sanctions, and Autonomous Smart Contracts

Antonopoulos, Andreas M., and Gavin Wood. 2018. Mastering Ethereum: Building Smart Contracts and DApps. Sebastopol, CA: O’Reilly Media.

Buterin, Vitalik. 2014. “A Next-Generation Smart Contract and Decentralized Application Platform.” Ethereum White Paper. Ethereum.

Office of Foreign Assets Control. “Questions on Virtual Currency.” United States Department of the Treasury. OFAC FAQs.

Office of Foreign Assets Control. 2021. Sanctions Compliance Guidance for the Virtual Currency Industry. United States Department of the Treasury. OFAC Guidance.

United States Department of the Treasury. 2022. “U.S. Treasury Sanctions Notorious Virtual Currency Mixer Tornado Cash.” August 8, 2022. Treasury Department.

United States Department of the Treasury. 2025. “Tornado Cash Delisting.” March 21, 2025. Treasury Department.

Van Loon v. Department of the Treasury, 122 F.4th 549, U.S. Court of Appeals for the Fifth Circuit, November 26, 2024. Fifth Circuit opinion.

Werbach, Kevin. 2018. The Blockchain and the New Architecture of Trust. Cambridge, MA: MIT Press.

Werbach, Kevin, and Nicolas Cornell. 2017. “Contracts Ex Machina.” Duke Law Journal 67: 313–382. Duke Law Journal.

Wood, Gavin. 2014. “Ethereum: A Secure Decentralised Generalised Transaction Ledger.” Ethereum Yellow Paper. Ethereum Yellow Paper.

13.10 Software Exploits, Adversarial AI, and Machine-Interpreted Inputs

National Institute of Standards and Technology. 2008. Technical Guide to Information Security Testing and Assessment. NIST Special Publication 800-115. NIST.

National Institute of Standards and Technology. 2025. Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations. NIST AI 100-2 E2025. NIST.

OWASP Foundation. “OWASP Top Ten.” OWASP.

Saltzer, Jerome H., and Michael D. Schroeder. 1975. “The Protection of Information in Computer Systems.” Proceedings of the IEEE 63(9): 1278–1308. DOI: 10.1109/PROC.1975.9939.

Zou, Andy, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. 2023. “Universal and Transferable Adversarial Attacks on Aligned Language Models.” arXiv:2307.15043.

13.11 Synthetic Biology, Nucleic-Acid Screening, and Biological Information Hazards

Department of Health and Human Services. 2023. Screening Framework Guidance for Providers and Users of Synthetic Nucleic Acids. Federal Register 88: 71507–71510. Federal Register.

Department of Health and Human Services. “Synthetic Nucleic Acid Security and Biorisk Management.” Administration for Strategic Preparedness and Response. HHS.

National Academies of Sciences, Engineering, and Medicine. 2018. Biodefense in the Age of Synthetic Biology. Washington, DC: National Academies Press. DOI: 10.17226/24890.

National Science Advisory Board for Biosecurity. 2023. Proposed Biosecurity Oversight Framework for the Future of Science. Bethesda, MD: National Institutes of Health. NIH.

Office of Science and Technology Policy. 2024. Framework for Nucleic Acid Synthesis Screening. Executive Office of the President. HHS-hosted framework.

Sharkey, Christopher M., Diane E. DiEuliis, Sonia Ben Ouagrham-Gormley, and others. 2024. “An Updated Framework for Synthetic Nucleic Acid Screening.” Health Security 22(4). PubMed Central.

World Health Organization. 2022. Global Guidance Framework for the Responsible Use of the Life Sciences: Mitigating Biorisks and Governing Dual-Use Research. Geneva: World Health Organization. WHO.

13.12 Cryptographic Disruption, Post-Quantum Migration, and Information Hazards

Bostrom, Nick. 2011. “Information Hazards: A Typology of Potential Harms from Knowledge.” Review of Contemporary Philosophy 10: 44–79. PDF.

Micheli, Chris, and others. 2023. Migration to Post-Quantum Cryptography: Preparation for Considering the Implementation and Adoption of Quantum Safe Cryptography. NIST Cybersecurity White Paper. NIST.

Mosca, Michele. 2018. “Cybersecurity in an Era with Quantum Computers: Will We Be Ready?” IEEE Security and Privacy 16(5): 38–41. DOI: 10.1109/MSP.2018.3761723.

National Institute of Standards and Technology. 2024. Module-Lattice-Based Key-Encapsulation Mechanism Standard. FIPS 203. NIST.

National Institute of Standards and Technology. 2024. Module-Lattice-Based Digital Signature Standard. FIPS 204. NIST.

National Institute of Standards and Technology. 2024. Stateless Hash-Based Digital Signature Standard. FIPS 205. NIST.

National Institute of Standards and Technology. 2024. “NIST Releases First Three Finalized Post-Quantum Encryption Standards.” August 13, 2024. NIST.

Shor, Peter W. 1997. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer.” SIAM Journal on Computing 26(5): 1484–1509. DOI: 10.1137/S0097539795293172.

13.13 Computability, Classification Limits, and the General Theory of Illegal Numbers

Floridi, Luciano. 2011. The Philosophy of Information. Oxford: Oxford University Press.

Gillespie, Tarleton. 2018. Custodians of the Internet: Platforms, Content Moderation, and the Hidden Decisions That Shape Social Media. New Haven, CT: Yale University Press.

Lessig, Lawrence. 1999. Code and Other Laws of Cyberspace. New York: Basic Books.

Rice, Henry Gordon. 1953. “Classes of Recursively Enumerable Sets and Their Decision Problems.” Transactions of the American Mathematical Society 74(2): 358–366. American Mathematical Society.

Turing, Alan M. 1936. “On Computable Numbers, with an Application to the Entscheidungsproblem.” Proceedings of the London Mathematical Society, second series, 42: 230–265. DOI: 10.1112/plms/s2-42.1.230.

Volokh, Eugene. 2005. “Crime-Facilitating Speech.” Stanford Law Review 57: 1095–1222. Stanford Law Review.

Wu, Tim. 2003. “When Code Isn’t Law.” Virginia Law Review 89(4): 679–751. JSTOR.

Zittrain, Jonathan. 2008. The Future of the Internet—and How to Stop It. New Haven, CT: Yale University Press. Harvard University.

The author has published over 5,500 AI-assisted papers at: https://www.researchgate.net/profile/Douglas-Youvan/research . Google Scholar has indexed these preprints, and if you want a particular reference, the AI mode of a Google search (Gemini) has efficiently catalogued these preprints.